

---

## Master thesis : Radar Detection for Drones

**Auteur** : Shehabi, Ammar

**Promoteur(s)** : Redouté, Jean-Michel; 12799

**Faculté** : Faculté des Sciences appliquées

**Diplôme** : Master : ingénieur civil électricien, à finalité spécialisée en "electronic systems and devices"

**Année académique** : 2021-2022

**URI/URL** : <http://hdl.handle.net/2268.2/14447>

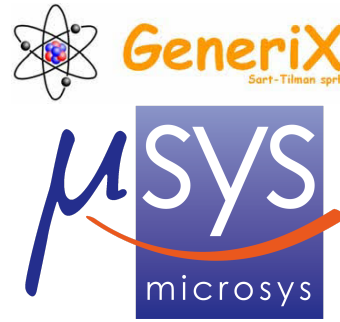
---

### *Avertissement à l'attention des usagers :*

*Tous les documents placés en accès ouvert sur le site le site MatheO sont protégés par le droit d'auteur. Conformément aux principes énoncés par la "Budapest Open Access Initiative"(BOAI, 2002), l'utilisateur du site peut lire, télécharger, copier, transmettre, imprimer, chercher ou faire un lien vers le texte intégral de ces documents, les disséquer pour les indexer, s'en servir de données pour un logiciel, ou s'en servir à toute autre fin légale (ou prévue par la réglementation relative au droit d'auteur). Toute utilisation du document à des fins commerciales est strictement interdite.*

*Par ailleurs, l'utilisateur s'engage à respecter les droits moraux de l'auteur, principalement le droit à l'intégrité de l'oeuvre et le droit de paternité et ce dans toute utilisation que l'utilisateur entreprend. Ainsi, à titre d'exemple, lorsqu'il reproduira un document par extrait ou dans son intégralité, l'utilisateur citera de manière complète les sources telles que mentionnées ci-dessus. Toute utilisation non explicitement autorisée ci-avant (telle que par exemple, la modification du document ou son résumé) nécessite l'autorisation préalable et expresse des auteurs ou de leurs ayants droit.*

---



End-of-study work carried out with a view to obtaining  
the master's degree in Electrical Engineering by  
SEHABI Ammar

---

## **Radar detection for drones**

---

*Academic promoter:* Prof. Jean-Michel REDOUTE  
*Industrial promoter:* Ir. Christophe GREFFE, GeneriX

University of Liège - Faculty of Applied Sciences

Academic year 2021 - 2022



## **Acknowledgements**

First of all, thanks to all people who gave me this opportunity to do this project and write this report. I would like to thank Prof. Jean-Michel Redoute who provided me with all kinds of help, support, and remarks during the research. I would like also to thank Mr. Christophe Greffe for welcoming me from the beginning, giving me many topics to choose and looking at my progress to guide me on many points.

Then big thanks to all micro-sys lab staff for being nice to me. Morgan Diepart for following me step by step and giving me many recommendations and suggestions to solve a lot of problems, Gabriel DiGregorio who gave a hand in designing the wooden setup and his tips with the final data set, and finally Herve Pierre and Francois Piron who have always been there for me and for my questions.

I am grateful also to Mr. Angel Calderon Gimenez in CEDIA hall who provided a great aid with designing the wooden setup for the device and made it an easy-to-use system, without forgetting to give me access to CEDIA to do my tests and lending me several materials.

Finally, my big thanks to all my family members for being such exceptional people and supporting me every single day without letting thousands of kilometers between us give up supporting me in my whole master's program.





# Contents

<b>Abbreviations</b>	<b>vii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 AWR1642 ODS EValuation Module (EVM) Features: . . . . .	2
<b>2 FMCW Radar concepts</b>	<b>5</b>
2.1 Introduction . . . . .	5
2.2 FMCW Radar . . . . .	5
2.2.1 Range resolution in a radar . . . . .	7
2.2.2 Maximum detected range . . . . .	8
2.2.3 Speed estimation of an object . . . . .	9
2.2.4 Doppler FFT . . . . .	11
2.2.5 Speed resolution . . . . .	12
2.2.6 Angle of Arrival estimation AoA . . . . .	12
2.2.7 Angle FFT . . . . .	16
2.2.8 Angle resolution . . . . .	16
<b>3 Chirp programming</b>	<b>17</b>
3.1 Chirp structure . . . . .	17
3.1.1 Impact of chirp programming on system . . . . .	18
3.1.2 Range parameters . . . . .	19
3.1.3 Speed parameters . . . . .	20
3.1.4 Angular parameters . . . . .	20
3.2 Chirp configuration . . . . .	20
3.2.1 Advanced chirp configuration . . . . .	21
3.3 Lab-SSR Source code . . . . .	23
3.3.1 USRR-20m Profile . . . . .	25
3.3.2 Derived parameters USRR-20m . . . . .	26
3.3.3 SRR-80m Profile . . . . .	29
3.3.4 Derived parameters - SRR-80m . . . . .	30
<b>4 Profile design</b>	<b>33</b>
4.1 Profile structure . . . . .	33
4.1.1 Timing parameters limitations: . . . . .	34
4.1.2 A valid profile example . . . . .	34
4.1.3 Configuration commands . . . . .	36
4.1.4 Followed methodology . . . . .	39
4.2 Programming the device using PuTTY . . . . .	41
4.3 mmWave Sensing Estimator . . . . .	42

<b>5 Experiments and Data analysis</b>	<b>45</b>
5.1 Drone detection . . . . .	45
5.2 A car moving perpendicularly to the detection axis of the radar . . . . .	49
5.3 Metallic and non metallic pieces detection . . . . .	54
5.4 Building detection . . . . .	58
5.5 A car moving in parallel to the detection axis of the radar . . . . .	60
<b>6 Conclusion and future work</b>	<b>63</b>
<b>Appendices</b>	<b>65</b>
<b>A Hardware setup</b>	<b>67</b>
<b>B profile125</b>	<b>69</b>
B.1 Profile 125m . . . . .	69
B.1.1 Derived parameters - SRR-125m . . . . .	70
<b>C Data set</b>	<b>73</b>
<b>D Technical issues</b>	<b>77</b>
<b>Bibliography</b>	<b>80</b>

# List of Figures

1.1	Front side of AWR1642 ODS EValuation Module. . . . .	3
1.2	Rear side of AWR1642 ODS EValuation Module. . . . .	4
2.1	Basic block diagram of FMCW radar. . . . .	6
2.2	Chirp and IF signals. . . . .	6
2.3	Amplitude-time chirp and IF signals. . . . .	7
2.4	Resolving process of two detected objects. . . . .	8
2.5	IF signal Phase. . . . .	10
2.6	Speed estimation. . . . .	11
2.7	Frame structure and Doppler FFT. . . . .	11
2.8	Speed resolving of two objects. . . . .	12
2.9	Additional travelled distance for second RX. . . . .	13
2.10	Angle of Arrival estimation. . . . .	14
2.11	2D-FFT matrix. . . . .	14
2.12	AoA Measurement accuracy w.r.t. angle. . . . .	15
2.13	Angle FFT. . . . .	16
3.1	Typical chirp signal associated with all configurable parameters. . . . .	17
3.2	Typical frame structure. . . . .	18
3.3	Transmitted chirp profile in LRR (125 m). . . . .	21
3.4	Structure of sub-frame in advanced configuration with three bursts looped twice. . . . .	22
3.5	Two sub-frames in advanced frame configuration profile. . . . .	22
3.6	Three modes of operation chirp profile. . . . .	23
3.7	Single advanced frame SRR(80m) profile. . . . .	24
3.8	Frame sequence structure of SRR(80m). . . . .	25
3.9	Single sub-frame of USRR-20m mode. . . . .	25
3.10	Single sub-frame for SRR-80m mode. . . . .	29
4.1	Chirp and profile allocation memory. . . . .	33
4.2	Snapshot of profile USRR20 .cfg file. . . . .	36
4.3	Snapshot of profile USRR20 .cfg file . . . . .	37
4.4	Successful device programming using PuTTY terminal emulator. . . . .	41
4.5	Snapshot on PuTTY terminal emulator while device is running. . . . .	42
4.6	A non valid estimated profile parameters. . . . .	43
4.7	A valid estimated profile parameters. . . . .	44
5.1	Tested environment and DJI drone model. . . . .	45
5.2	Snapshot of experiment output. . . . .	46
5.3	Visibility elevation FOV of AWR1642 EVM w.r.t. the drone. . . . .	46

5.4	Detection at height of 2 m. . . . .	48
5.5	Detection at height of 3 m. . . . .	49
5.6	Detection at height of 4 m. . . . .	49
5.7	Detection at height of 5 m. . . . .	49
5.8	Detection at height of 6 m. . . . .	49
5.9	Car position in location 1 and location 2. . . . .	50
5.10	Detected pointed polar plot at location 1. . . . .	52
5.11	Detected pointed polar plot at location 2. . . . .	53
5.12	Two cars along longitudinal axis. . . . .	53
5.13	Range and noise profile. . . . .	54
5.14	CPU load. . . . .	54
5.15	Maximum theoretical detected azimuth range for different angles. . . . .	55
5.16	Non detected two metallic objects. . . . .	55
5.17	Four detected objects at different distances. . . . .	57
5.18	Detected Plastic dustbin at distance about 3m and AoA about 20°. . . . .	57
5.19	Non detected plastic dustbin and detected Cartoon box. . . . .	57
5.20	Musée en plein air and B52 buildings. . . . .	59
5.21	Invisible detection of a part of the building. . . . .	59
5.22	Visible detection of the B52 building. . . . .	59
5.23	The tested location 3 in the Science Park. . . . .	60
5.24	Location3, moving car at 50 Km/h. . . . .	61
5.25	Location3, moving car at 30 Km/h. . . . .	62
5.26	Location3, moving car at 70 Km/h. . . . .	62
A.1	Complete setup along with all components fixed. . . . .	68
A.2	Printed PCB radar holder. . . . .	68
A.3	Web camera to film test session. . . . .	68
A.4	Battery cohesive to the wooden floor. . . . .	68
A.5	Battery LiPo and converter fixed to the setup. . . . .	68

# List of Tables

1	Table of abbreviations . . . . .	ix
3.1	RCS typical values for some objects. . . . .	19
3.2	Radar range equations' parameters. . . . .	20
3.3	Configured valid profile example for LRR(125m). . . . .	21
3.4	A valid Advanced configuration profile example. . . . .	23
3.5	Two operation modes of chirp. . . . .	24
3.6	USRR-20m profile and chirp parameters. . . . .	26
3.7	USRR-20m Sub-frame parameters. . . . .	26
3.8	SRR-80m profile and chirp parameters. . . . .	29
3.9	SRR-80m Sub-frame parameters. . . . .	30
4.1	Timing parameters threshold values. . . . .	34
4.2	Channel configuration. . . . .	34
4.3	ADC configuration. . . . .	35
4.4	ADC buffer configuration. . . . .	35
4.5	Profile configuration 1. . . . .	35
4.6	Profile configuration 2. . . . .	35
4.7	Chirp configuration. . . . .	35
4.8	Frame configuration. . . . .	35
4.9	BPM configuration. . . . .	35
4.10	GUI configuration. . . . .	36
4.11	CFAR configuration. . . . .	36
4.12	configuration file .cfg explanation. . . . .	38
5.1	Drone detection at height 2 m and 3 m. . . . .	47
5.2	Drone detection at height 4 m and 5 m. . . . .	47
5.3	Drone detection at height 6 m and threshold angle values. . . . .	48
5.4	Location 1, car perpendicular to detection axis. . . . .	50
5.5	Location 2, car perpendicular to detection axis. . . . .	51
5.6	Detection of Aluminum panel when angle $\Theta$ is $0^\circ$ and $20^\circ$ respectively. . . . .	56
5.7	Detection of Aluminum panel when angle $\Theta$ is $40^\circ$ and $55^\circ$ respectively. . . . .	56
5.8	Detection of a building covered with metallic material (B52 building) . . . . .	58
5.9	Detection of a building not covered with metallic material (Musée en plein air). . . . .	58
5.10	Location 3, car parallel to detection axis of the radar. . . . .	60
B.1	SRR-125m profile and chirp parameters. . . . .	69
B.2	LRR-125m Sub-frame parameters . . . . .	69
C.1	A portion of the obtained data set during location 3 test . . . . .	75







# Abbreviations

<b>Abbreviation</b>	<b>Description</b>
ADC	Analog Digital Converter
AoA	Angle of Arrival
API	Application Programming Interface
BPM	Binary Phase Modulation
CFAR	Constant False Alarm Rate
CW	Continuous Wave
CCS	Code Composer Studio
DSP	Digital Signal Processing
DSS	Digital Sub System
2D-FFT	Two Dimensions Fast Fourier Transform
EKF	Extended Kalman Filter
EVM	EValuation Module
FFT	Fast Fourier Transform
FMCW	Frequency Modulated Continuous Wave
FOV	Field Of View
GUI	Graphical User Interface
IF	Intermediate Frequency
IFF	If and only If
KF	Kalman Filter
LPF	Low Pass Filter
LRR	Long Range Radar
MCU	Micro Controller Unit
MIMO	Multiple Inputs Multiple Outputs
MMIC	Monolithic Microwave Integrated Circuit
MRR	Medium Range Radar
MSS	Master Sub System
NF	Noise Figure

<b>Abbreviation</b>	<b>Description</b>
RADAR	Radio Detection And Ranging
RCS	Radar Cross Section
RF	Radio Frequency
RRE	Radar Range Equation
RX	Receiver antenna
SDK	Software Development Kit
SNR	Signal to Noise Ratio
SOP	Sense-On-Power
SRR	Short Range Radar
TDM	Time Division Multiplexing
TI	Texas Instruments
TLV	Type Length Value
TX	Transmitter antenna
UART	Universal Asynchronous Receiver-Transmitter
USRR	Ultra Short Range Radar
✓	Detected
×	Not Detected

Table 1: Table of abbreviations



# Chapter 1

## Introduction

Drone applications are more and more common and keep growing, particularly for logistics or for aerial inspection (eg windmill). The project was about to implement a mmWave radar sensor to automatize these applications, the main purpose of this sensor is obstacle avoidance (other drones, buildings). Moreover, to exploit the advantages of such system over camera systems that could not be as efficient as radar ones by night or in a foggy environment. The purpose of the research is to study the use of the AWR1642 Obstacle Detection Sensor Evaluation Module and discover its real capabilities to validate its use for drone applications.

At the very beginning of the research, an automotive mmWave low power radar system module was chosen among several ones based on some characteristics such as (maximum range and speed, resolution, power consumption, and price...).

Different experiments were performed with the radar using self-made support to hold the radar safely, in order to understand the radar in practice and verify its capability in open and closed environments against static and dynamic objects. By doing that it was possible to have an initial clue about the system performance.

Then a deep dive into the internal software structure of the radar was done to be able to program it and improve some aspects, it was, therefore, possible to configure the device with the desired application for the user.

Finally, with the new programmed configurations that were made previously and compared with different results that were in hand already, it was possible to make a data set and conclude the research such that we have a clear idea about the real performance of the device and how much we can exploit its capabilities in our application, as it was clear enough if we can use it for drone applications or not.

In this introduction chapter (**chapter 1**), an organization of the report will be given and a brief explanation for each chapter is provided. Then, a very quick conclusion about the work that has been done during the project with an overview of the device that was tested in this thesis AWR1642BOOST-ODS.

In **chapter 2**, the concept of FMCW radar from a theoretical point of view is presented, besides the important concepts that are parts of any detection system such as range, speed, and angular resolution with corresponding mathematical formulas are also provided and explained. These concepts will be the basis of later chapters.

After showing what is FMCW radar in chapter 2 and what kind of signals it transmits, **chapter 3** explains the structure of the transmitted signal and how it is linked with many configurable

parameters that impact the performance of the radar. A deep dive into the internal software structure of the device along with the interpretation of the source code into clear data tables and mathematical formulas are provided to understand what kind of signal has been sent at each time radar is operated..

The work has been done in chapter 3 was helpful in understanding how the device thinks and knowing if it is possible to design our own signal and configure the device with. Therefore, **chapter 4** was about that, where it could be possible to make our first signal and cover a higher range than what the device was programmed with, it was achievable also to make another kind of signaling profile where two, three, or four signals could be sent simultaneously for different application.

After testing the designed signal and making different tests and experiments for many dynamic and static objects providing that several materials were under test as well, **chapter 5** comes to show some of summarizing data set that was made based on the extracted and concluded information from experiments in both closed and open environments. We summarized and interpreted this data set into binary and polar plots w.r.t. to each corresponding test.

The last chapter is **chapter 6** to conclude and decide the validity of working on such a device for drone applications, it discusses how far we could use it in automotive applications as well. Propositions of potential space for development and future work on what has been done in this thesis or could be done separately are provided.

The research concludes that the chosen mmWave radar module is far away to be an implemented system on the drone or to be used for drone detection and it is not that efficient in such an application. However, the device proved that it is efficient in automotive applications detection for static and dynamic objects, and it provides large flexibility for the development part as well.

## 1.1 AWR1642 ODS Evaluation Module (EVM) Features:

The device manufactured by Texas Instruments TI integrates a 77–81-GHz FMCW single chip, it comprises a millimeter-wave (mmWave) radio-frequency (RF) and analog baseband signal chain for two transmitters (TX) and four receivers (RX), as well as two customer-programmable processors in the form of a C674x Digital Signal Processor (DSP) and an ARM Cortex-R4F Micro Controller Unit (MCU). It provides by default the following features :

- Frequency band : 77-81 GHz.
- The device comprises the two transmitters (TX) and four receivers (RX), two processor cores, first one DSS (DSP) for advanced signal processing and second one MSS controls the overall operations of the device.
- Transmit power: 12 dBm.
- Maximum range: by default it is 80 m for cars and trucks and 20 m for smaller objects.
- Range resolution : by default it is 36 cm when detection up to 80 m and 4.3 cm when detection up to 20 m.

- Maximum unambiguous relative speed : by default it is  $\pm 56$  Km/h when detection up to 80 m and  $\pm 18$  Km/h when detection up to 20 m.
- Speed resolution : 1.87 Km/s.
- ADC sampling rate up to 6.25 MSamp/s .
- Field of View (FOV) covers  $\pm 70^\circ$  in azimuth plane and  $\pm 20^\circ$  in elevation plane.
- Price : 299 \$.
- Power consumption for the AWR1642 chip is within [1.38 - 2.14] Watt.

Figures 1.1 and 1.2 show the top and rear sides of AWR1642 ODS EVM respectively :

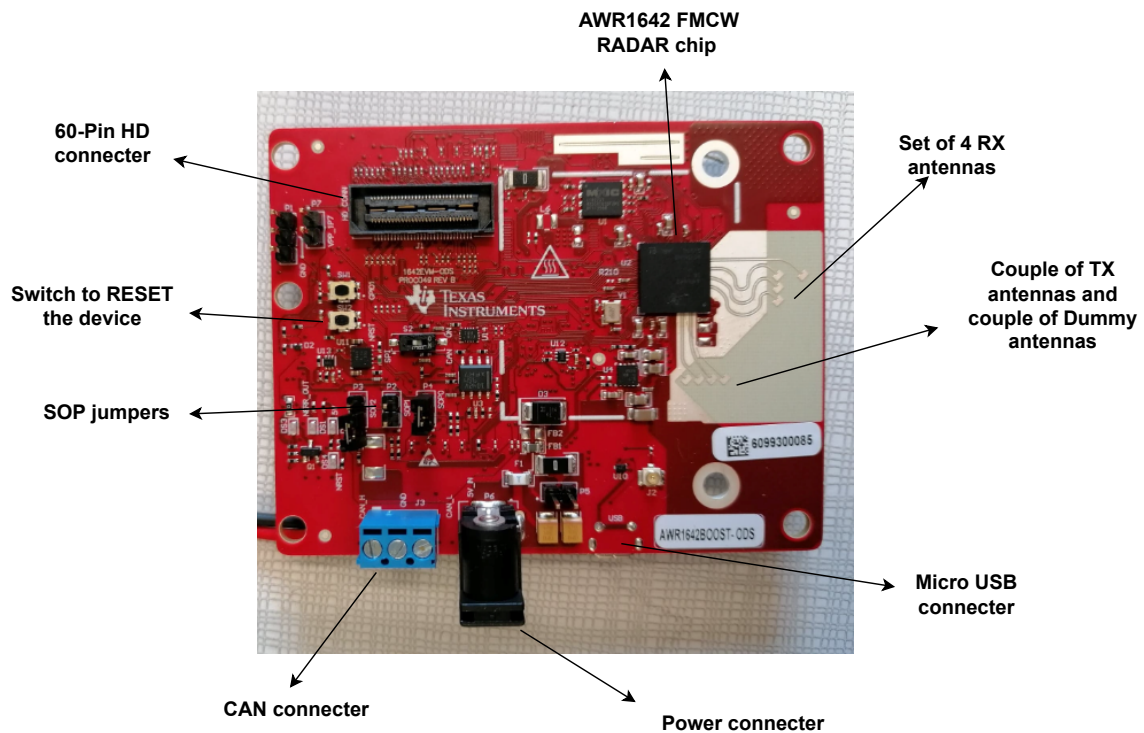


Figure 1.1: Front side of AWR1642 ODS Evaluation Module.

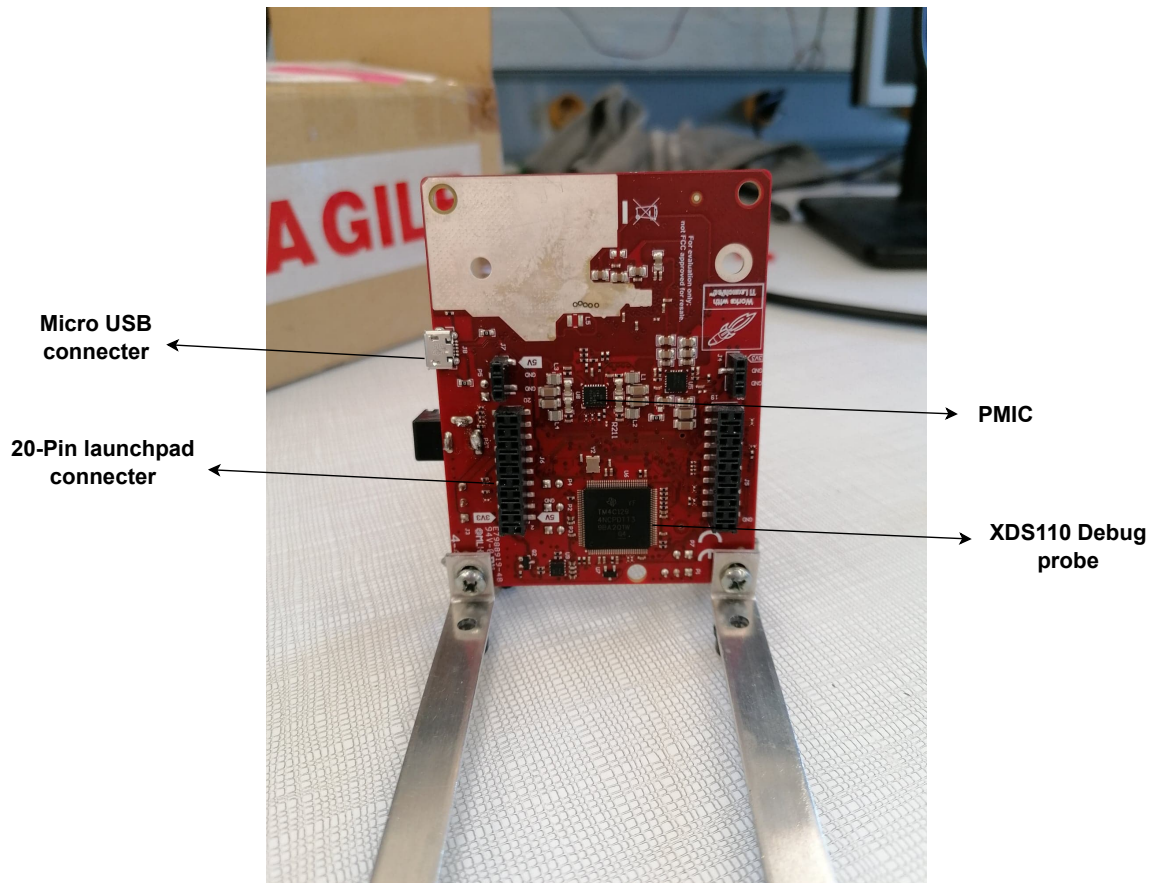


Figure 1.2: Rear side of AWR1642 ODS Evaluation Module.

# Chapter 2

## FMCW Radar concepts

### 2.1 Introduction

In this chapter, a global overview of the working principle of Frequency Modulated Continuous Wave FMCW radar from a theoretical and mathematical point of view will be presented, as it will be provided some information about the device AWR1642 Evaluation Module EVM. We will explain some of its characteristics, how they can influence sensor performance and what are the factors that are involved.

### 2.2 FMCW Radar

It is a special type of radar sensors that transmits a unique signal called a 'chirp' whose frequency increases linearly with time and that is why called frequency-modulated radar. In contrast to Continuous Wave CW radar, FMCW radar can change its operating frequency or phase during the measurement at a known rate over a defined time period. Possibilities of radar measurements through runtime measurements are only technically possible with these changes in the frequency (or phase) where the reflected frequency signal is received by the radar and compared. The difference between the transmitted and reflected signals is directly proportional to the time of flight, which in turn is therefore proportional to the range. Working principle is illustrated in the general block diagram of FMCW radar system in figure 2.1 :

- A synthesizer generates a chirp which is transmitted by TX antenna.
- The chirp is reflected by an object and the reflected chirp is received at the RX antenna at a delayed time.
- The RX signal and TX signal are 'mixed', the resulting signal is called an IF signal (Intermediate Frequency) and it has a constant frequency. A single object in front of the radar produces an IF signal.
- Further digitization, filtering and processing operations are applied on IF signal by Digital Signal Processing DSP unit.



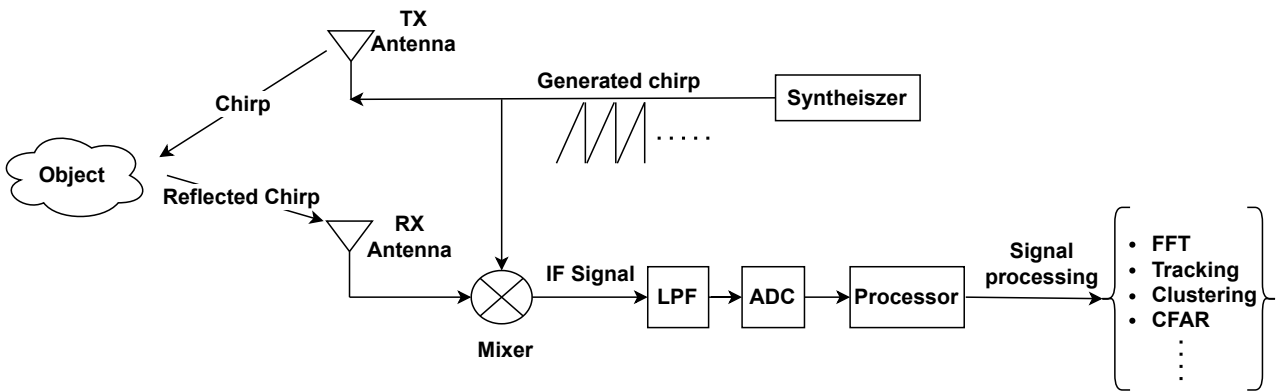


Figure 2.1: Basic block diagram of FMCW radar.

Figure 2.2 shows the amplitude-time transmitted signal (chirp) on the top-left and frequency-time on the bottom-left, it is a sinusoidal signal that starts with a specific frequency which increases constantly at a given rate within a maximum bandwidth range of 4 GHz (for AWR1642 device),  $S$  is the slope of chirp which indicates the rate of chirp at which ramping up is performed (in this figure chirp starts at 77 GHz and stops at 81 GHz with bandwidth  $B = 4$  GHz) and time period  $T_c = 40 \mu s$ , so it results in a slope of  $100 \text{ MHz}/\mu s$ . Both  $S$  and  $B$  are important parameters for radar performance.

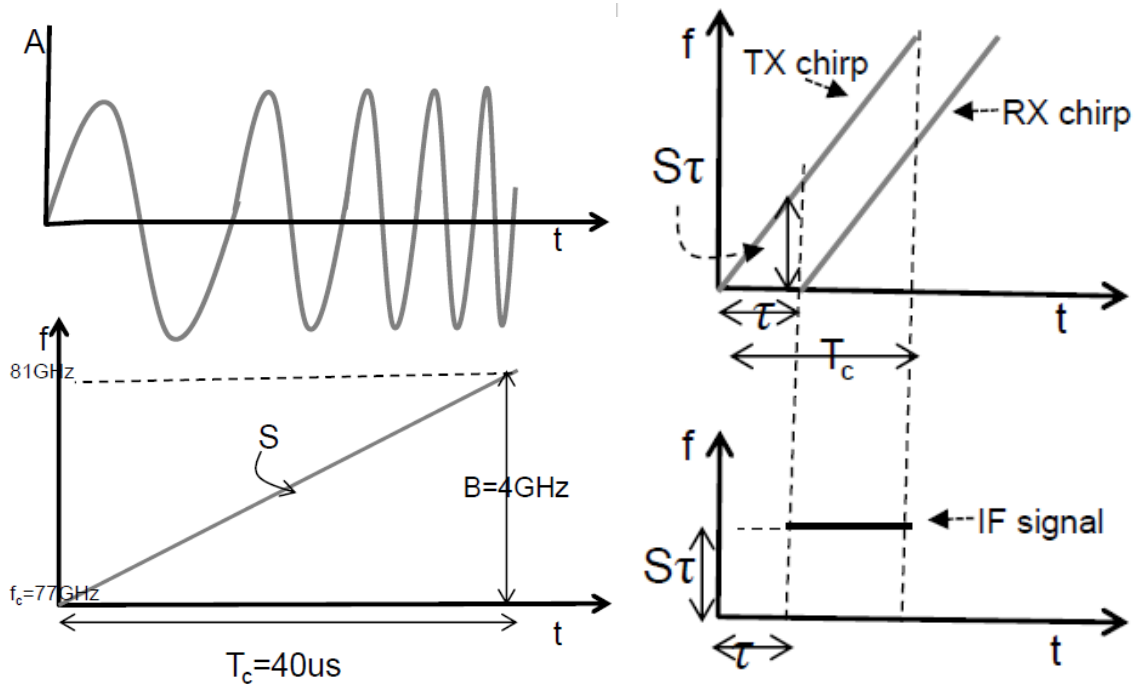


Figure 2.2: Chirp and IF signals.

On the top right of figure 2.2 transmitted and received chirps in the frequency-time plot where we notice that RX chirp is a delayed version of TX chirp by round-trip time  $\tau$ , on the bottom right we see the mixed resultant IF signal, it starts exactly at the same time of receiving RX chirp and it has a constant frequency equal to :

$$f = S \times \tau = S \times \frac{2d}{c} \quad (2.1)$$

Where  $f$  is IF frequency (Hz),  $S$  denotes the slope of the chirp (MHz/ $\mu$ s),  $\tau$  is round-trip time (s),  $d$  is distance of object (m) and  $c$  is the speed of light (m/s).

### 2.2.1 Range resolution in a radar

It refers to the ability of the radar how it can display two or more close objects clearly in the frequency domain, in other word it is the smallest distance between two objects that allows them to be detected as separate objects rather than one. Range resolution of FMCW radar is given by :

$$d_{Res} = \frac{c}{2B} \quad (2.2)$$

Where  $d_{Res}$  is range resolution (m),  $c$  is speed of light (m/s) and  $B$  is the bandwidth of the chirp (Hz). Best range resolution that can be obtained using our radar module AWR1642 is equal to 4 cm when the maximum sweep bandwidth 4 GHz is reached.

After the IF signal is generated, it has multiple tones and the frequency of each one corresponds to a distance  $d$  of an object, then an analog to digital conversion process is performed on it, and finally, an FFT will be performed on the output of ADC resulting multiple peaks in frequency spectrum where each peak refers to the range of a detected object. The entire previous process is called 'Range FFT' in which we have the best estimation value of the range of the detected object.

Figure 2.3 shows the TX chirp and two reflected RX chirps from two objects on the top of the plot, on the bottom we see the amplitude-time plot for two IF signals (red and green) corresponding to both objects, they have very close frequencies during the current time window, if we resolve the current signals in frequency domain then the result will be an ambiguous because we will have only one peak in the frequency spectrum and we are not able to distinguish the range for each object.

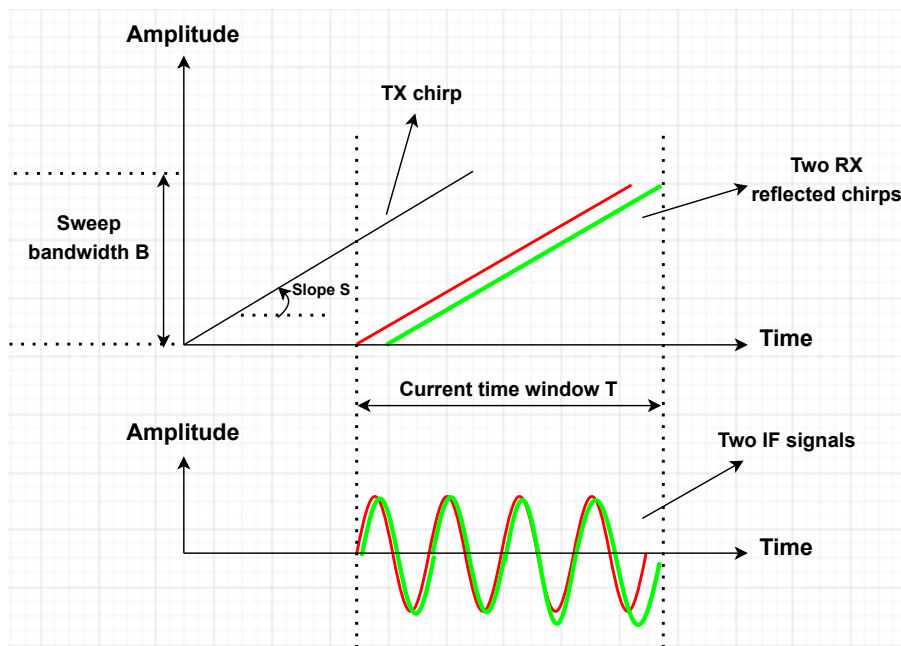


Figure 2.3: Amplitude-time chirp and IF signals.

A solution is shown in figure 2.4, is to make a longer observation period by increasing the length of the IF signal to  $2T$ , in other word, increase chirp duration, this, in turn, increases the bandwidth to  $2B$  which improves the range resolution according to formula 2.2. In this case, we have two clear peaks in the frequency domain estimating the range of each object, and both objects are resolved.

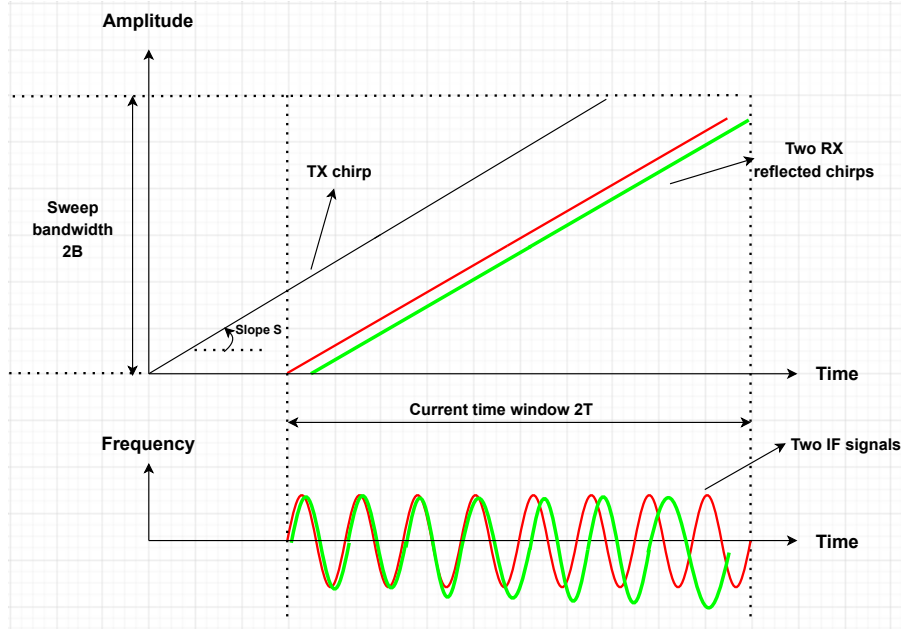
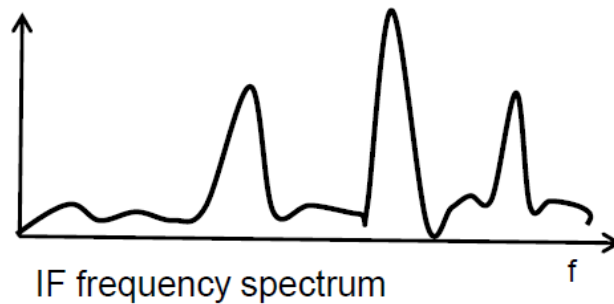


Figure 2.4: Resolving process of two detected objects.

Figure below shows a plot of successful resolving process done after performing range FFT in frequency spectrum for 3 objects. Location of each peak in the frequency spectrum is corresponding to the range of an object.



### 2.2.2 Maximum detected range

The maximum detected range of a radar is directly proportional to the sampling rate of its ADC. Let us consider a maximum detected range up to some  $d_{Max}$ , then according to formula 2.1 the maximum IF frequency will be equal to  $f_{Max} = \frac{2Sd_{Max}}{c}$  and consequently the bandwidth will be from 0 to  $f_{Max}$  therefore the sampling rate of ADC namely  $f_s$  must be higher or equal to  $f_{Max}$ :

$$f_s \geq \frac{2Sd_{Max}}{c}$$

$$d_{Max} = \frac{f_s c}{2S} \quad (2.3)$$

Or, maximum detected range can be written in term of the IF frequency as ( $f_{Max} = IF_{Max}$ ):

$$d_{Max} = \frac{f_{Max} c}{2S} \quad (2.4)$$

In practice and according to the source code and datasheet, two IF frequencies are used in detection process based on current sampling rate mode. However, only first case i.e (formula 2.5) will be considered. In complex 1x sampling mode :

$$IF_{Max} = 0.9 \times ADC_{Samp} \quad (2.5)$$

In complex 2x sampling mode :

$$IF_{Max} = 0.9 \times \frac{ADC_{Samp}}{2} \quad (2.6)$$

### 2.2.3 Speed estimation of an object

To measure the speed of a moving object in the FOV of the sensor we have to define the phase of the generated IF signal. The initial phase of the output signal at mixer output is the difference between the two phases of both inputs (transmitted and received chirps). If we consider an object at distance  $d$  and moving at speed of  $V$ , IF signal is sinusoid  $A \sin(2\pi f t + \phi_0)$  where the phase difference is given by :

$$\Delta\phi = 2\pi f_c \Delta\tau = 2\pi f_c \frac{2\Delta d}{c}$$

$$\Delta\phi = \frac{4\pi\Delta d}{\lambda} \quad (2.7)$$

$\lambda$  is the wavelength (m),  $\Delta\phi$  is the phase difference between the two inputs (degree) and  $\phi_0$  is the initial phase of IF signal which will be the basis of estimating the speed because it is very sensitive to small changes in object motion and this property will be exploited to estimate the actual speed. Note that the phase changes linearly with distance variations of the object.

Figure 2.5 shows the amplitude-time plot of the transmitted, received, and IF signals. Initially the phase  $\phi_0$  of the IF signal at C is equal to the phase difference of both transmitted and received chirps at A and B respectively. Let's consider an object that has a small displacement in the forward direction such that the change in the distance now is  $\Delta d$ , then the received chirp will be delayed a bit more and the transmitted chirp travels a bit more distance as well. Both received chirp and the corresponding IF signal are shifted by a small amount  $\Delta\tau$  (blue curves), in this case, the phase of the IF signal at F will be the phase difference between E and D, and the phase at E is the same as earlier (at B) but the phase at D is equal to the phase at A plus an additional amount.

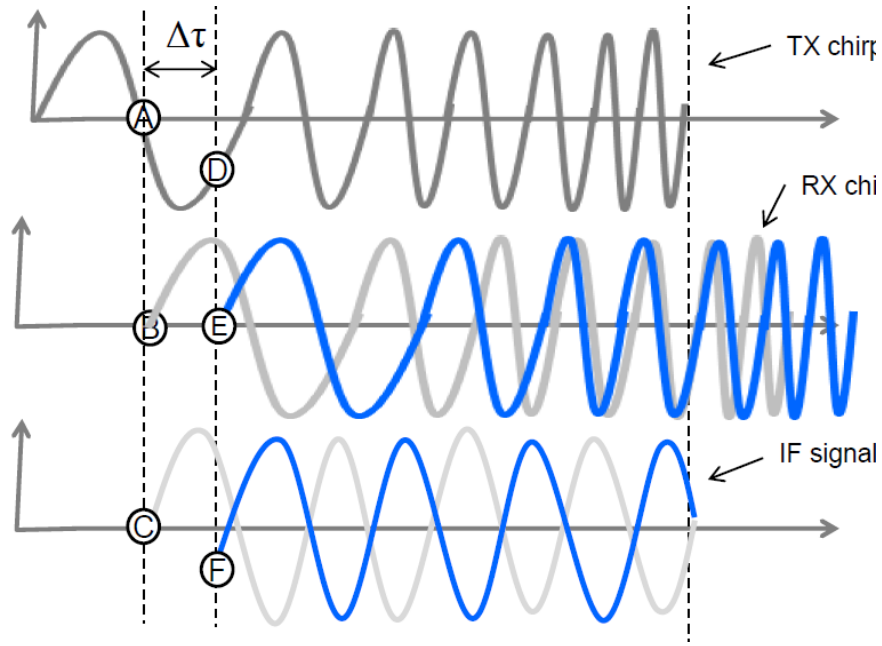


Figure 2.5: IF signal Phase.

Speed estimation is illustrated in figure 2.6:

- Transmit two sequencing chirps separated by time period  $T_c$ .
- The range-FFT corresponding to each chirp has peaks in the same location (i.e same frequency) but with different phase.
- The measured phase difference  $\omega$  corresponds to the motion of that object which moves at a speed of  $V$  during the time period  $T_c$ , i.e the object displaces by amount of  $d = VT_c$  is given by :

$$\omega = \frac{4\pi VT_c}{\lambda}$$

$$V = \frac{\lambda\omega}{4\pi T_c} \quad (2.8)$$

- There is a limit for speed measurement such that it will be unambiguous as long as the phase difference resides in  $[-180^\circ, 180^\circ]$ . Therefore, the maximum relative speed  $V_{Max}$  that can be measured by two sequencing chirps spaced  $T_c$  apart is given by:

$$|\omega| < \pi$$

$$\frac{4\pi VT_c}{\lambda} < \pi$$

$$V_{Max} = \frac{\lambda}{4T_c} \quad (2.9)$$

- Note that  $T_c$  is dependent on how fast the frequency sweep can be performed in other word the faster the ramp of the frequency the minimum  $T_c$  and therefore the higher maximum unambiguous speed, where the maximum allowed ramp-up in AWR1642 module is  $100 \text{ MHz}/\mu\text{s}$ .

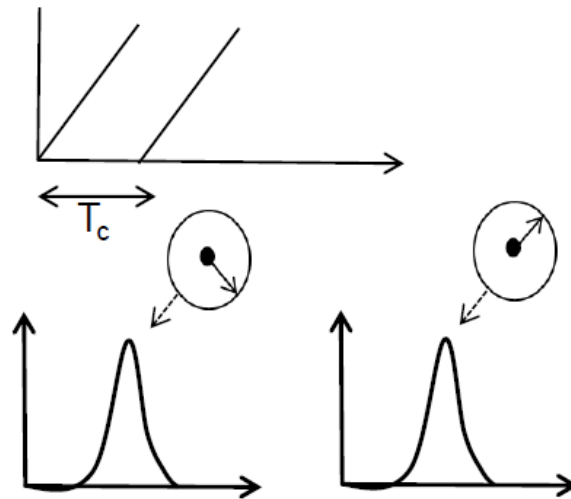


Figure 2.6: Speed estimation.

### 2.2.4 Doppler FFT

This technique is used when there are multiple objects at the same distance from the radar and have different speeds, in this case, the phasor of each peak in the frequency domain has two components corresponding to two objects for example. The simple phase comparison method does not work here, what can be done is basically to send a series of equi-spaced chirps instead of only two, this sequence is called a "Frame" it can also be called the unit of FMCW radar. A frame by definition is the timing template that we impose via inserting the sequence of the chirps that form the frame. Number of frames that need to be transmitted, number of defined chirps within a given frame and periodicity can be programmed as it will be explained later in chirp programming chapter, hence, the duty cycle of transmission operation. Figure 2.7 shows the frame structure:

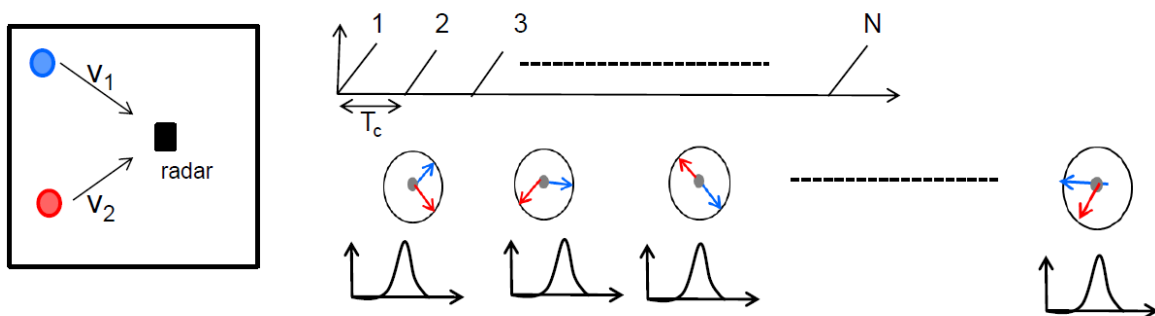


Figure 2.7: Frame structure and Doppler FFT.

Range FFT for these chirps has peaks at the same location, each peak has two rotated phasors  $\omega_1, \omega_2$  corresponding to the two speeds  $V_1, V_2$  respectively. Then FFT of these phasors results in two distinct angular frequency components which can be used to estimate the speed of two objects based on formula 2.8. Figure 2.8 shows the two angular frequencies corresponding to speed of two objects.

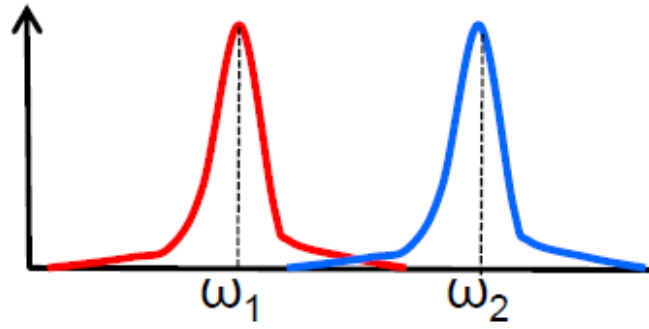


Figure 2.8: Speed resolving of two objects.

### 2.2.5 Speed resolution

It refers to the ability to separate out objects with small speed differences. It is the minimum speed value between two moving objects at speed  $V_1, V_2$  to be displayed clearly and shown as two distinct speeds (angular frequency). Speed resolution mostly depends on the transmit frame, it is given by:

$$V_{Res} = \frac{\lambda}{2T_f} \quad (2.10)$$

Where  $T_f = NT_c$  is the frame time (s),  $N$  is the number of sequencing chirps that being transmitted and  $T_c$  is total chirp time (s), which includes chirp time + idle time between two consecutive chirps.

$$T_c = T_{CD} + T_{IT} \quad (2.11)$$

where  $T_{CD}$  denotes chirp duration (ramp-end time) and  $T_{IT}$  denotes idle time, both of them are measured in ( $\mu$ s).

### 2.2.6 Angle of Arrival estimation AoA

It is the third dimension in the radar system after range and speed. As mentioned earlier, the IF signal is very sensitive to the small displacement of an object in front of the radar, the same concept is exploited here to estimate the Angle of Arrival of an object. For that at least two RX are needed, in figure 2.9 it can be seen that when a chirp is transmitted by TX, two rays are reflected from the object, one travels a distance 'd' and the other  $d + \Delta d$ . This additional distance  $\Delta d$  results in an additional phase and the overall phase in 2D-FFT (range and speed) will be changed. The phase difference of IF signal between two antennas is given by :

$$\omega = \frac{2\pi\Delta d}{\lambda} \quad (2.12)$$

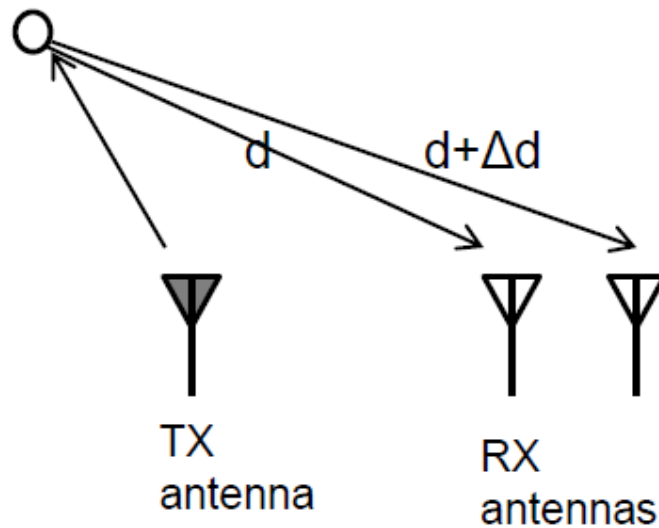


Figure 2.9: Additional travelled distance for second RX.

How we estimate AoA is shown in figure 2.10:

- TX antenna transmits a series of chirps.
- Assuming that we have a TX and two RX antennas, chirp is transmitted and reflected back from the object on the RX antennas, where  $\Theta$  (Theta) is the angle of arrival of the beam and 'd' is the distance between the two RX antennas. Beam at second RX travels higher distance than beam at first RX by  $d \sin(\Theta)$
- the first RX receives the reflected signal with an angle  $\Theta$  whereas the second RX receives the reflected signal with a delayed version where it is traveled an additional distance equal to  $d \sin \theta$ .
- 2D-FFT resolving matrix has peaks for each RX which refers to the range and speed of detected objects, these peaks are at the same frequency locations in the frequency spectrum but in different phases.
- Then, phase difference can be measured and exploited to estimate AoA which is give by :

$$\omega = \frac{2\pi d \sin(\Theta)}{\lambda}$$

$$\Theta = \arcsin\left(\frac{\lambda \omega}{2\pi d}\right) \quad (2.13)$$



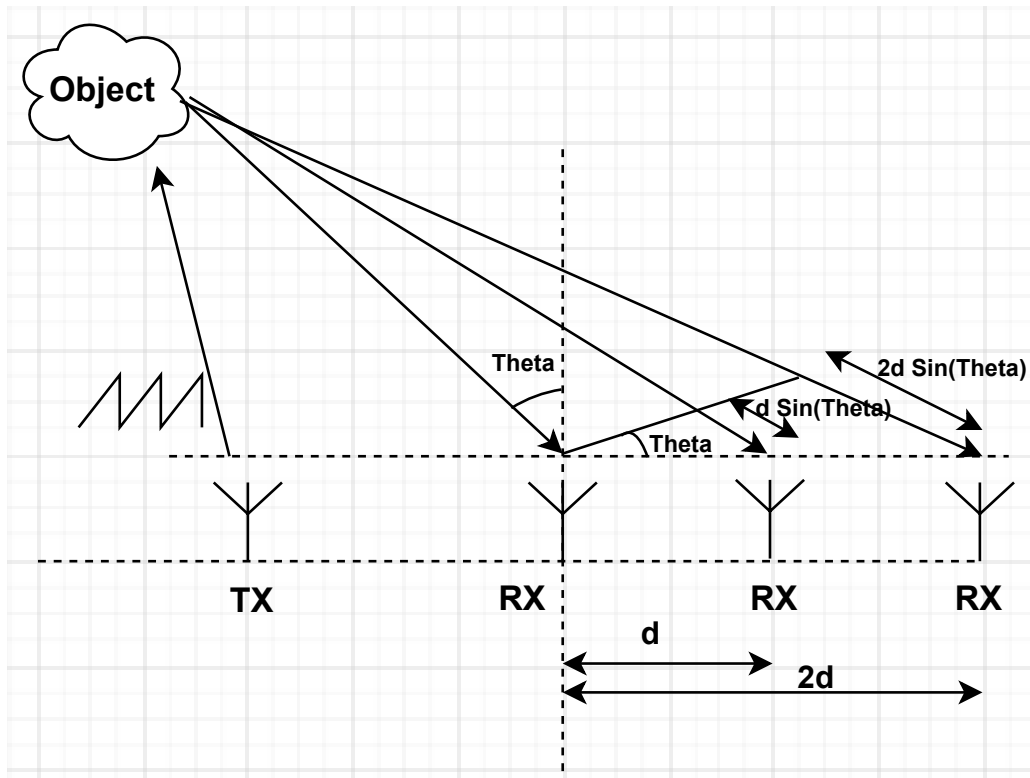


Figure 2.10: Angle of Arrival estimation.

- Figure 2.11 shows the 2D-FFT matrix corresponding to each RX, it can be seen that we have the same signal value for a given location and speed of an object but phasors are different.

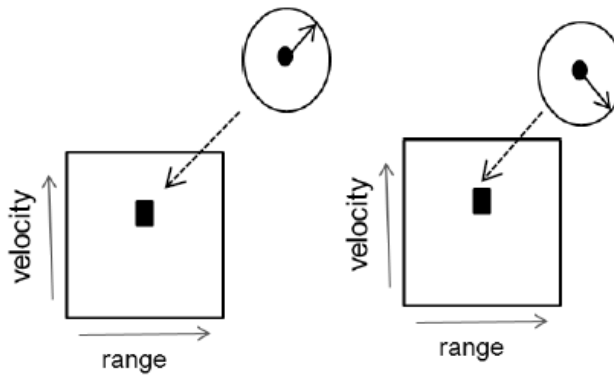


Figure 2.11: 2D-FFT matrix.

• **Field Of View (FOV)**

According to formula 2.13, it can be observed that the relationship between the measurement  $\Theta$  and phase  $\omega$  is non-linear which means that if  $\Theta = 0^\circ$ , then the accuracy of the measurement is the best and the higher  $\Theta$  is the less accuracy. That is shown in figure 2.12 where radar can have the best measurement when the object is directly in front of the sensor with  $\Theta = 0^\circ$ . Angle ranges in red are equal to  $70^\circ$  representing the theoretical field of view in both sides. Let us consider three potential detected objects in the vicinity of sensor object 1, object 2, and object 3, and they have different angles w.r.t. the sensor  $\Theta_1$ ,  $\Theta_2$  and

$\Theta_3$  respectively then each of them reflects a given amount of energy back to sensor except object 3 since it is out of FOV and receive a little amount and reflects back almost nothing. Object 1 has a higher probability than object 2 to be detected, if we have an object perpendicular to the sensor i.e  $\Theta = 0^\circ$  it is detected regardless of its material or surface up to a particular distance due to higher energy reflection back. In this figure, assume that the three objects have the same size, surface, and material as we neglect the angle of each object w.r.t. the elevation plan since it has an impact also on the detection process as will be seen in the data analysis chapter.

Same as in speed measurement, there is also a limit for measuring AoA for radar with a TX and two RX antennas such that this measurement will be unambiguous as long as the phase difference between two antennas resides in  $[-180^\circ, 180^\circ]$ . Therefore, angle  $\Theta_{Max}$  that can be measured by the radar is given by:

$$\begin{aligned} |\omega| &< \pi \\ \frac{2\pi d \sin(\Theta)}{\lambda} &< \pi \\ \Theta_{Max} &= \arcsin\left(\frac{\lambda}{2d}\right) \end{aligned} \quad (2.14)$$

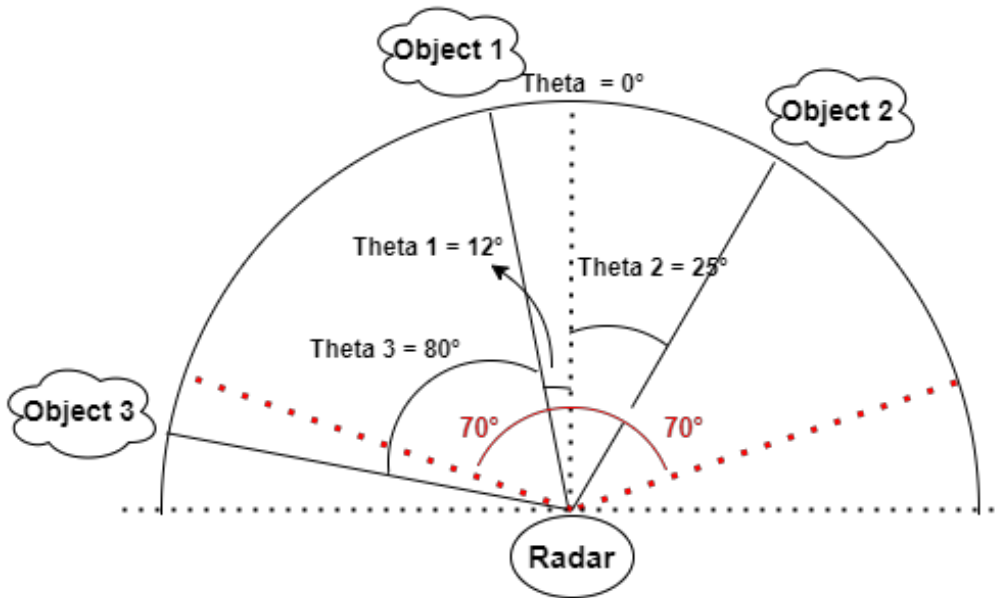


Figure 2.12: AoA Measurement accuracy w.r.t. angle.

Note that if there are two RX antennas spaced apart  $d = \frac{\lambda}{2}$ , then the biggest possible FOV =  $\pm 90^\circ$ . However, for AWR1642 EVM, the distance between two consecutive RX is 2.1 mm as was measured using ALTIUM DESIGNER tool in the micro sys lab, and the wavelength :

$$\lambda = \frac{c}{f} = \frac{3 \times 10^8 \text{ m/s}}{77 \times 10^9 \text{ Hz}} = 3.9 \text{ mm}$$

Thus the maximum theoretical FOV of device is:

$$\Theta_{Max} = \arcsin\left(\frac{3.9 \times 10^{-3} \text{ m}}{2 \times 2.1 \times 10^{-3} \text{ m}}\right) = 68.2^\circ$$

### 2.2.7 Angle FFT

Same analogy as Doppler FFT, this technique is used when we have two or more objects at the same range and speed w.r.t. radar. In the 2D-FFT matrix, there will be two peaks at the same location for both antennas without having a clear idea of how much the contribution of each object in this signal is.

The solution is to have more RX antennas up to N as shown in figure 2.13 such that each antennas has a peak in 2D-FFT matrix at same location, and a pair of rotated phasors correspond to the two detected objects. FFT of these phasors show up as two distinct signals in the frequency spectrum at  $\omega_1, \omega_2$  corresponding to the two objects as shown in figure 2.13, where  $\omega_1, \omega_2$  is the rate of rotation of both objects ( $\frac{\text{radian}}{\text{sample}}$ ), then AoA of both objects can be estimated using to formula 2.13 to obtain finally to frequency components as shown in figure 2.8.

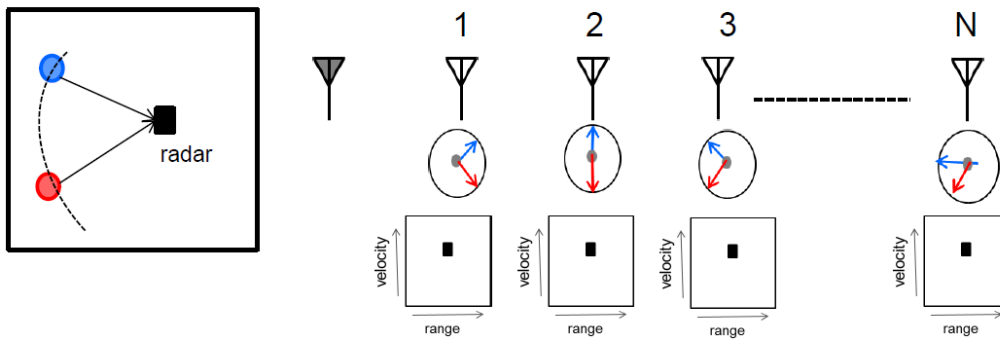


Figure 2.13: Angle FFT.

### 2.2.8 Angle resolution

It is the ability to resolve two objects close to each other by angle. Let two objects at angles of  $\Theta$  and  $\Theta + \Delta\Theta$  respectively to the radar, angle resolution denotes how small  $\Delta\Theta$  can be such that the radar can still see both objects as distinct objects and process them as two different peaks in the frequency spectrum. Angle resolution can be estimated by using the following formula:

$$\Theta_{Res} = \frac{\lambda}{Nd \cos \Theta} \tag{2.15}$$

Where  $\lambda$  is the wavelength (m), N is the number of RX antennas, d is the distance between two consecutive RX antennas (m) and  $\Theta$  is the angle of arrival (degrees). The term 'Nd' is called the length of the antenna array, there are 4 RX antennas in the device, and the distance between every two consecutive ones is 2.1 mm, therefore the total length is 8.4 mm. However, the length of the RX array is fixed and the only factor that remains to improves angular resolution is AoA, thus :

$$\downarrow \Theta \Rightarrow \uparrow \cos \Theta \Rightarrow \downarrow \Theta_{Res}$$

# Chapter 3

## Chirp programming

### 3.1 Chirp structure

It's one of the most important aspects that impacts the overall performance of the radar by choosing the right and the proper configurations for the radar module. We can choose whatever parameters of chirp depending on the application that we seek, SRR, MRR, LRR...etc. MMIC radar provides large flexibility in chirp programming where we can also configure the radar with multiple chirp parameters in a single-mode meaning that transmission of several kinds of the signal simultaneously and each of which is responsible for a particular detection mode such that radar displays detected objects far away and close by ones.

Figure 3.1 shows the typical chirp signal associated with different configurable parameters:

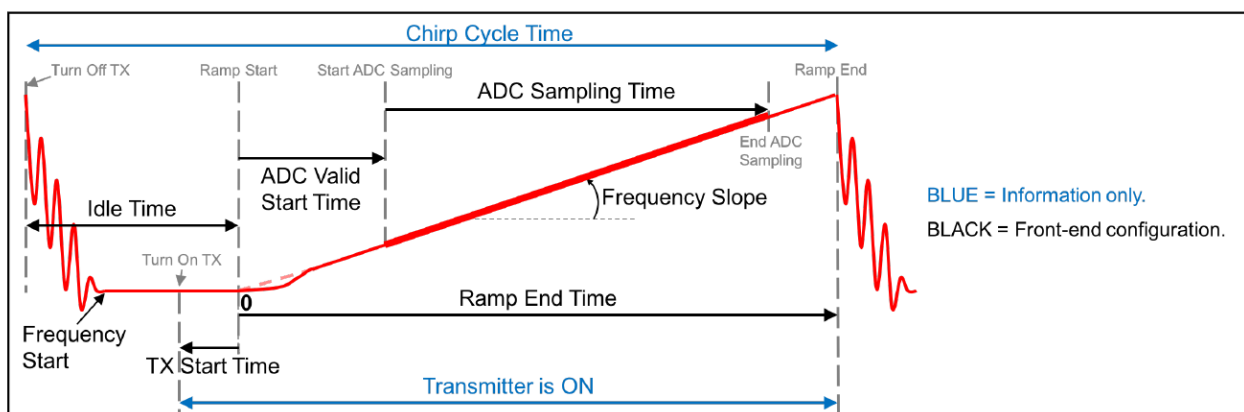


Figure 3.1: Typical chirp signal associated with all configurable parameters.

Figure 3.2 depicts the typical frame design, a frame by definition is the sequence of transmitted chirps followed by inter frame time, each frame is divided to multiple sub-frames as it will be explained later in this chapter. This represents 'Fast FMCW' modulation.

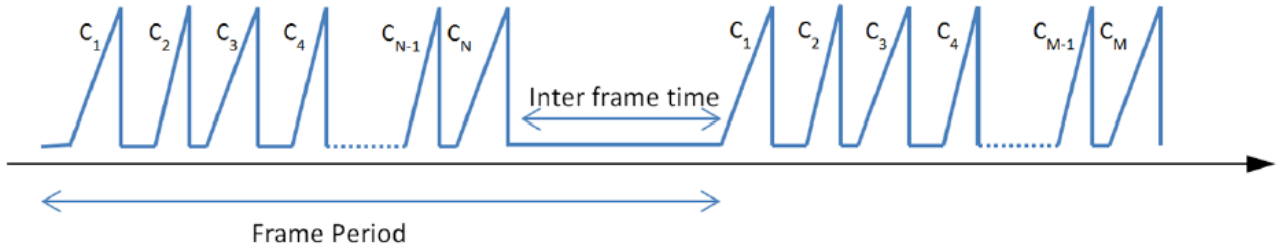


Figure 3.2: Typical frame structure.

### 3.1.1 Impact of chirp programming on system

During the transmission of a series of chirps, this sequence forms a frame, this frame can be used as an observation window. The most important configurable parameters of chirp are, start frequency, frequency slope, idle time, ADC start time, and ramp end time. They form all together the key point to manipulate system performance, and they are typically considered in any radar application.

- **Start frequency (Hz):**

It represents at which frequency the device starts transmitting chirps, mmWave radars TI has several operation frequency bands. For the device in hand, it supports maximum bandwidth up to 4 GHz within [77-81] GHz.

- **Frequency slope (MHz/ $\mu$ s):**

It defines the rate at which chirp sweeps within the frequency band, the device supports a maximum slope up to 100 MHz/ $\mu$ s. It has a direct impact on the maximum detected range and an indirect impact on speed parameters. Based on formula 2.4 :

$$\downarrow Freq_{Slope} \Rightarrow \uparrow Range_{Max}$$

- **Idle time ( $\mu$ s):**

It defines the time between the end of the previous chirp and start of the next chirp, it has a direct impact on speed parameters. Based on formulas 2.8 and 2.10 :

$$\downarrow Idle_{Time} \Rightarrow \uparrow Speed_{Max} \Rightarrow \uparrow Speed_{Res}$$

- **ADC start time ( $\mu$ s):**

The time from the start of the ramp when the ADC starts sampling the data, it has a direct impact on speed parameters. Based on formulas 2.8 and 2.10 :

$$\downarrow ADC_{StartTime} \Rightarrow \uparrow Speed_{Max} \Rightarrow \uparrow Speed_{Res}$$

- **Ramp end time (chirp duration  $\mu$ s):**

It is the time from the start of the ramp until the chirp continues ramping. After this time, the synthesizer frequency is reset to the start frequency of the next chirp. It is a critical parameter because it impacts strongly both range and speed parameters.

$$\downarrow Ramp_{end} \Rightarrow \uparrow Speed_{Para} \Rightarrow \downarrow Range_{Para}$$

### 3.1.2 Range parameters

There are two parameters of interest, maximum detection range, and range resolution. When an application in the automotive domain is desired, it is important to detect objects as far as possible but that is not achieved in practice as it will be shown later on in this chapter and the data analysis chapter where a trade-off must be considered. They are governed by frequency slope and sweep bandwidth Based on formulas 2.2 and 2.4.

There is another important factor that can affect the maximum detection range, Radar Cross Section RCS which represents how much each object is detected by the radar such that how much it reflects energy back to the radar. There is none a general formula that describes RCS because it depends on many parameters such as the geometrical shape of the object, its material nature (conducting or not), orientation, surface, and the frequency in some cases. The higher the RCS, the better detection, in general, it can be described as :

$$\sigma = \text{Object Cross Section} \times \text{Object Directivity} \times \text{Object Reflectivity} \quad (3.1)$$

Where  $\sigma$  is RCS ( $m^2$ ), it is almost impossible to measure RCS of objects with ordinary tools since it needs a special measurement lab and tools, table 3.1 shows some typical values of RCS for some objects, these values were estimated by the manufacturer and were mentioned in the documentation of the device, the higher RCS the better detection.

Detected Object	RCS value [ $m^2$ ]
Truck	100
Car	5
Motorcycle	3.2
Adult	1
child	0.5

Table 3.1: RCS typical values for some objects.

In previous chapter it was not mentioned another way to determine the maximum detected range since this way is a bit sophisticated and relies on many other factors such as SNR value:

- RF performance of the radar device  $\rightarrow$  TX output power, RX noise figure, and idle frame time ( $NT_c$ ).
- Antenna parameters  $\rightarrow$  TX and RX antenna gain.
- Object characteristics  $\rightarrow$  RCS value.
- Minimum SNR required by the detection algorithm to detect an object.

Equation 3.2 is called Radar Range Equation (RRE) and can be used to determine the maximum range based on SNR value :

$$Range_{Max} = \sqrt[4]{\frac{P_t \times G_{RX} \times G_{TX} \times c^2 \times \sigma \times NT_c}{f_c^2 \times (4\pi)^3 \times KT \times NF \times SNR_{Min}}} \quad (3.2)$$

Table 3.2 shows the value of each parameter in RRE and if we substitute the total frame time  $T_f = NT_c$  based on formula 2.11 with value of 6 ms for example which corresponds to a maximum range of 20 m, we get approximately the same value. Therefore, as it could be seen there are many mathematics calculation are involved in this formula and that is why it is better to consider only equation 2.4 when it comes to find the maximum range.

Symbol	Parameter involved	Corresponding value
$P_t$	TX output power	12 dBm
$G_{RX}$	RX gain	24 dB typically
$G_{TX}$	TX gain	dB typically
$\sigma$	Radar Cross Section	5 $m^2$ for car
$NT_c$	total frame time	6 ms (for example)
$f_c^2$	frequency start	77 GHz
$KT$	Boltzmann constant	$4.11 \times 10^{-21}$ Jeul at 298 k
$NF$	noise figure	15 dB
$SNR_{Min}$	minimum SNR programmed	15 dB

Table 3.2: Radar range equations' parameters.

### 3.1.3 Speed parameters

As range parameters, there are two parameters involved and they are basically related to the timing parameters of the chirp. They are expressed in formulas 2.9 and 2.10. For speed resolution, it depends on how many chirps are transmitted per single frame, i.e frame period = number of transmitted chirps multiplied by chirp period, the longer the frame period the better speed resolution. We can distinguish between two frame periods, the first one is the idle frame period (including idle time), and the second one is the active frame period (without idle time). This aspect has little effect on speed parameters during programming the device.

### 3.1.4 Angular parameters

As previously, we have a maximum angular range that can be detected and angular resolution. The maximum unambiguous angular range denotes the field Of View FOV that can radar detects objects in. Estimation Angle of Arrival AoA of an object is related to delayed versions of each received signal at each RX which results in a phase shift due to the distance  $d$  between two consecutive RX, this phase shift is equal to  $\frac{2\pi d \sin\theta}{\lambda}$  and can be exploited to estimate the angle  $\theta$ . They are expressed in equations 2.13 and 2.15.

## 3.2 Chirp configuration

One of the most interesting features that AWR1642 radar module implements is that it allows users to configure various editions of chirps in a single signal profile. **Chirp profile** is a timing template where it could differentiate the different parameters among defined chirps such as (slope, start frequency, idle time...etc). It is possible to have up to 4 profiles, and up to 512 unique chirps can be pre-programmed and stored in chirp RAM, each chirp definition can belong to one of the 4 profiles. **A frame** is the sequence of chirps that is created by looping over these chirps for a specific number of loops up to 255 times.

Table 3.3 shows an example of a programmed profile for long distance up to 125 m, and the transmitted signal is shown in the figure 3.3 which is contained in a single frame such that radar transmits infinite number of frames :

Parameter	Unit	Corresponding value
Maximum unambiguous range	m	125
Sweep bandwidth	MHz	183
Ramp slope	MHz/ $\mu$ s	6.5
Idle time	$\mu$ s	2
Number of chips/frame	-	96
Range resolution	m	0.88
Chirp duration	$\mu$ s	30
Maximum unambiguous speed	Km/h	54.72
ADC sampling rate(1X)	Ks/s	6000
Maximum beat frequency (90 % of ADC sampling rate)	KHz	5400
Number of samples/chirp	-	156
Idle frame time	ms	3.07
Active frame time	ms	2.88
Range FFT memory size	KB	512

Table 3.3: Configured valid profile example for LRR(125m).

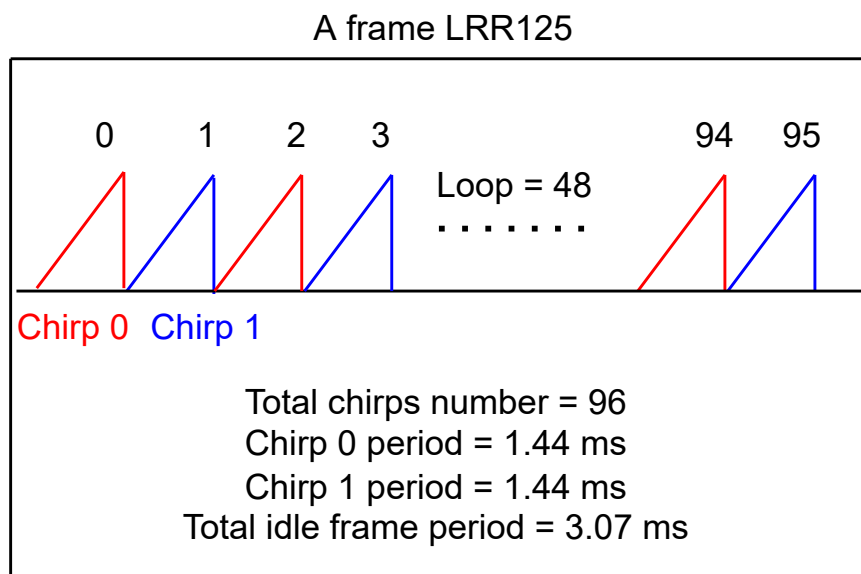


Figure 3.3: Transmitted chirp profile in LRR (125 m).

### 3.2.1 Advanced chirp configuration

It is an implemented feature in AWR1642 radar which supports multiple detection modes at a time, e.g SRR with LRR simultaneously using a so-called 'advanced frame configuration'. A frame can be seen as a sequence of chirps and composed of sub-frames (up to 4) and each of which represents a detection mode. Using the provided API `rlSetAdvFrameCfg`, it is possible to break the frame into sub-frames and define different chirps as needed. Each sub-frame contains a burst of chirps (up to 512) which are associated with one of the four profiles, these bursts can be even



looped up to **64** times. Figure 3.4 shows the structure of the frame and sub-frame:

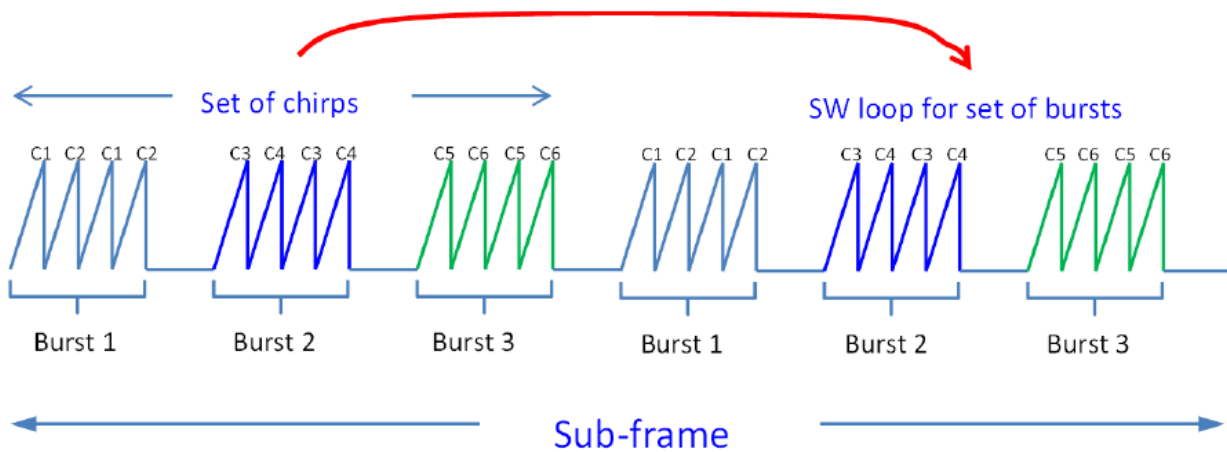


Figure 3.4: Structure of sub-frame in advanced configuration with three bursts looped twice.

There is a margin of flexibility since the device allows users to make a different set of chirps in each sub-frame as it can be seen in figure 3.5 , and could possibly have different antenna configurations for each set of chirps. In this mode of operation there are some restrictions that we must respect due to time windowing conditions:

- Inter-burst time should be  $\geq 50 \mu s$
- Inter sub-frame time should be  $\geq 100 \mu s$
- Inter frame time should be  $\geq 200 \mu s$ .

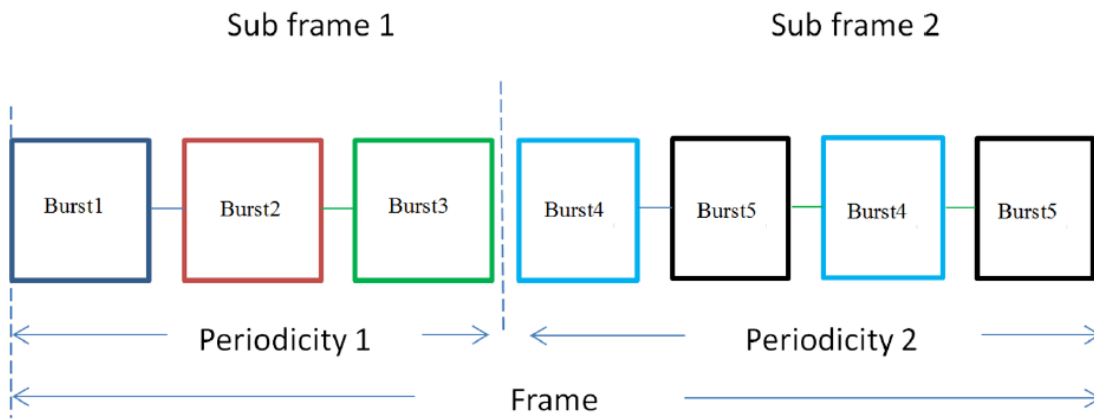


Figure 3.5: Two sub-frames in advanced frame configuration profile.

Table 3.4 shows a programmed valid example of advanced profile contains three sub-frames used to cover objects up to 7.5 m with high range resolution and objects up to 75 m and 150 m with poorer range resolution. Figure 3.6 show the corresponding transmitted chirping profile.

Parameter	Unit	Profile 0	Profile 1	Profile 2
Sub-frame ID	-	0	1	2
Maximum unambiguous range	m	7.5	75	150
Sweep bandwidth	MHz	3940	580	225
Ramp slope	MHz/ $\mu$ s	80	10	5
Idle time	$\mu$ s	2	2	5
Number of chips/frame	-	32	32	128
Range resolution	m	0.04	0.26	0.75
Chirp duration	$\mu$ s	50	60	45
Maximum unambiguous speed	Km/h	30.06	26.85	36
ADC sampling rate(1X)	Ks/s	5000	6250	6250
Number of samples/chirp	-	240	360	250
Idle frame time	ms	1.79	2.11	6.4
Active frame time	ms	1.6	1.92	5.76
Range FFT memory size	KB	512	512	1024

Table 3.4: A valid Advanced configuration profile example.

### Three subframes configuration

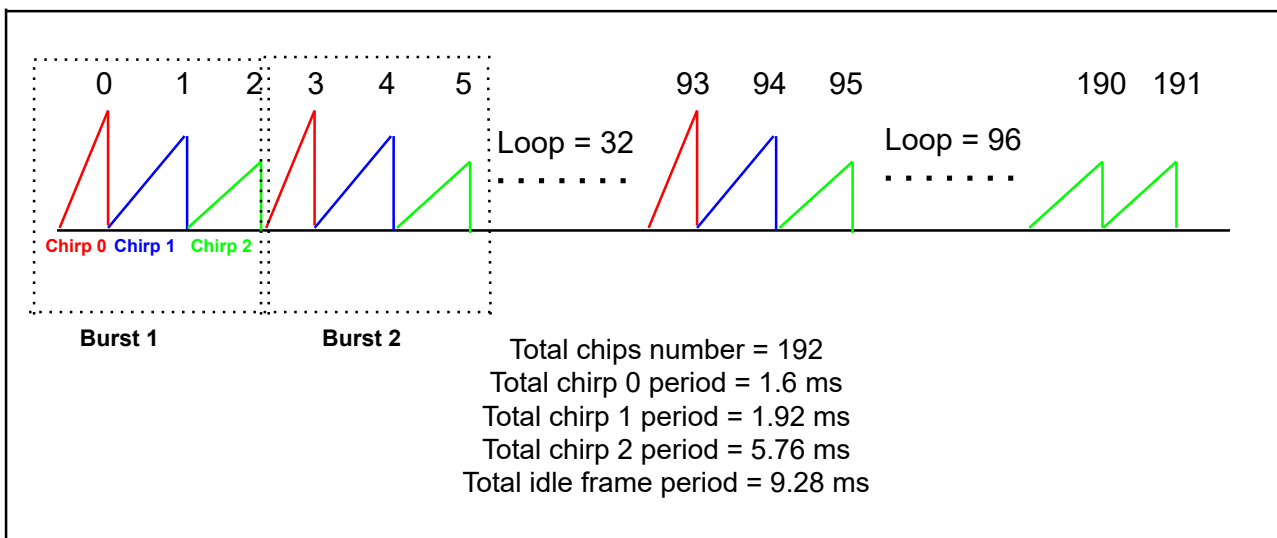


Figure 3.6: Three modes of operation chirp profile.

## 3.3 Lab-SSR Source code

In this section an explanation of the internal software structure of SSR lab will be presented, we are going to explain it and interpret the source code in terms of configuration part into mathematical formulas and data tables to be able to understand how is configured, in later sections a development of transmitted signal is presented to improve detection process.

SRR lab enables the estimation and tracking of the position (in the azimuthal plane) and velocity of objects in its field of view up to 80m, traveling as fast as 55 km/h. The AWR1642 is configured to be a multi-mode radar, meaning that, while it tracks objects at 80 m with poor range resolution, it can also generate a rich point cloud of objects at 20 m for close-by objects with high range resolution, so that both objects (car) at high distance and smaller obstacles (human) close-by can be detected. Table 3.5 illustrates the two operation modes of AWR1642 module.

Operation mode	Maximum range	Range resolution	Maximum speed
Short Range Radar SRR-80m	high range $\approx 80$ m	low range resolution $\approx 30$ cm	high speed $\approx 55$ km/h
Ultra Short Range Radar USRR-20m	low range $\approx 20$ m	high range resolution $\approx 4.5$ cm	low speed $\approx 18$ km/h

Table 3.5: Two operation modes of chirp.

Since radar operates in two modes essentially two sub-frames per frame, with alternate sub-frames belonging to one of the two modes. But the maximum speed of LRR mode is not enough so a new technique is used to improve it where two kinds of chirps are created with a maximum unambiguous velocity of 55 km/h and the other having a max unambiguous velocity of  $(55/1.2)$  km/h, then use the Chinese' remainder theorem' to estimate the true velocity which results an overall maximum velocity of 90 km/h. Figure 3.7 shows the exact transmitted chirp within a given profile.

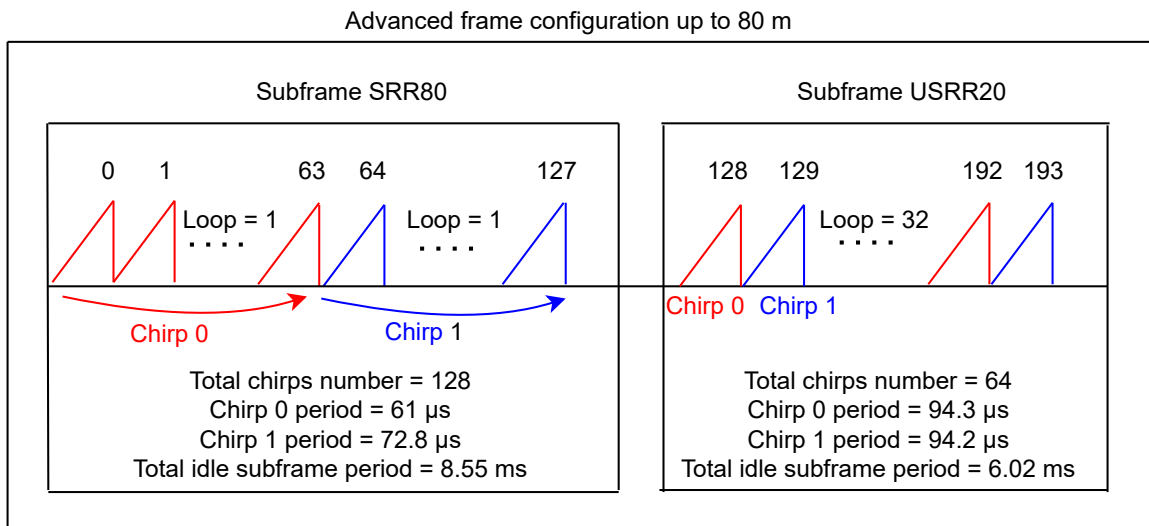


Figure 3.7: Single advanced frame SRR(80m) profile.

Where numbers 0,1,2....193 represent the index of each chirp in the memory. After understanding the transmission of the chirp procedure and the structure of each signal we can move to explain every single profile with its parameters including the derivation of range and speed parameters. Radar sends an infinite number of frames which will be as figure 3.8 shows:

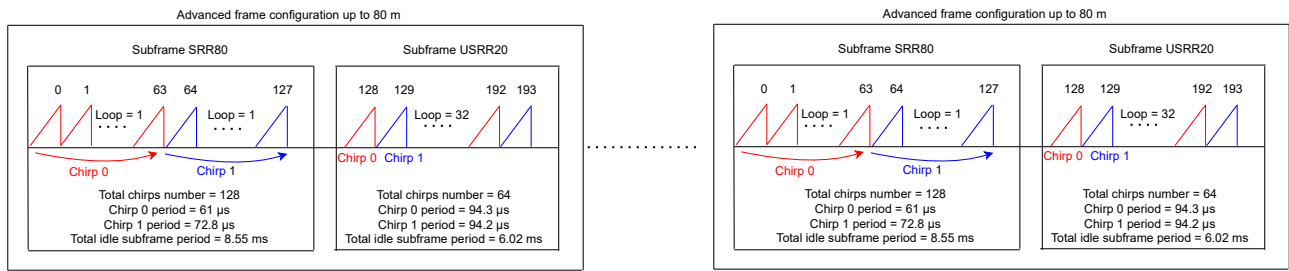


Figure 3.8: Frame sequence structure of SRR(80m).

### 3.3.1 USRR-20m Profile

This profile can be used when the detection of small objects at small distances is desired or when a measurement of low range resolution is wanted. This is a profile with a very high range resolution of 4.3 cm, low range detection of 20 m, a maximum speed of 18 km/h, and angular resolution of 15°. Transmitted signal in such a profile is as figure 3.9 :

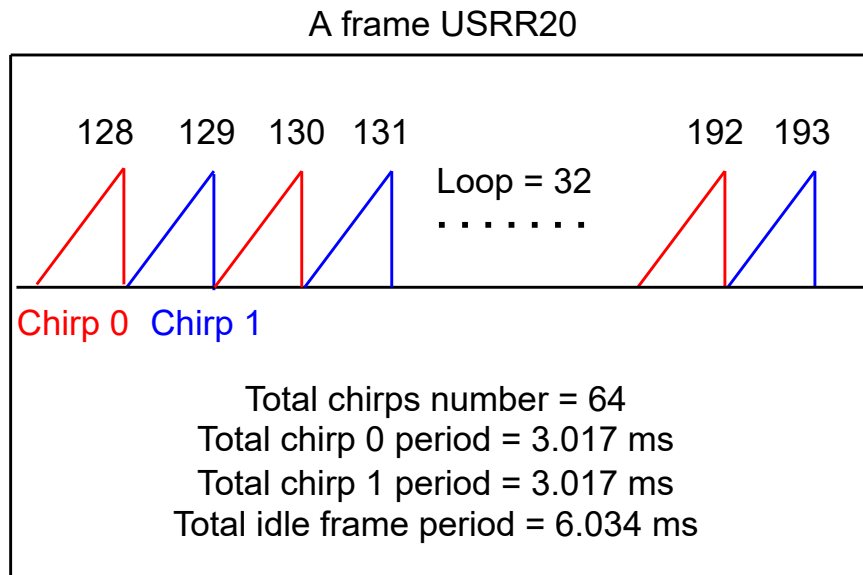


Figure 3.9: Single sub-frame of USRR-20m mode.

In every profile, there are three sets of parameters important for the detection. They are profile, chirp, and frame parameters. during the programming of the device with any desired signal, special care should be considered to the interaction between these parameters and a match must be met otherwise we have errors or not proper detection as we will see later on. The three set of parameters of profile, chirp, and frame are shown in tables 3.6(a), 3.6(b) and 3.7 respectively. The corresponding pseudo-code of USRR-20m profile is shown in code 1.

Profile ID	1
Sampling rate	6222 KSamp/s
Number of ADC samples	512
Start frequency	77 GHz
Frequency slope	42 MHz/s
Idle time	7 $\mu$ s
Ramp end time	87.28 $\mu$ s
TX start time	1 $\mu$ s
ADC start time	4.8 $\mu$ s

((a)) USRR-20m Profile parameters.

	Chirp 0	Chirp 1
Profile ID	1	1
Start index	128	129
End index	128	129
Start frequency variation	0	0
Frequency slope variation	0	0
Idle time variation	0	0
ADC start time variation	0	0
TX antenna	1	2

((b)) USRR-20m Chirp parameters.

Table 3.6: USRR-20m profile and chirp parameters.

Start index	128
End index	129
Loop count	32
Periodicity	30 ms
Number of real ADC samples	$512 \times 2 = 1024$
Number of complex ADC samples	512
Chirp 0 number	$1 \times 32 = 32$
Chirp 1 number	$1 \times 32 = 32$
Total chirp number	$2 \times 32 = 64$
Transmitters number	2
Number of angle bins	32

Table 3.7: USRR-20m Sub-frame parameters.

```

1 initialize_profile_parameters ();
  initialize_chirp0_parameters ();
  initialize_chirp1_parameters ();
  initialize_frame_parameters ();

6 compute_maximum_range ();
  compute_range_resolution ();
  compute_period_chirp0 ();
  compute_period_chirp1 ();
  compute_maximum_speed ();
11 compute_speed_resolution ();

```

Listing 1: Pseudo-code configuration of the USRR-20m profile.

### 3.3.2 Derived parameters USRR-20m

Deriving parameters of USRR-20m profile is done by extracting all information from previous tables and calculate each parameter using equations from previous chapter :

- **Maximum detected range :**

Using equations 2.4 and 2.5 , maximum detected range can be found as follows:

$$Range_{Max} = \frac{IF_{Max} \times c}{2S} = \frac{0.9 \times SamplingRate \times c}{2S} \quad (3.3)$$

$$Range_{Max} = \frac{0.9 \times 6222 \times 10^3 (Samp/s) \times 3 \times 10^8 (m/s)}{2 \times \frac{42 \times 10^6 Hz}{10^{-6} s}}$$

$$Range_{Max} = 19.99 \text{ m}$$

- **Range resolution :**

Another form of formula is derived rather than formula 2.2 , we start from standard formula until the formula that has been used in the source code:

$$Range_{Res} = \frac{c}{2B} = \frac{c}{2ST_c}$$

Where S is the slope (MHz/ $\mu$ s) and equal to :

$$S = \frac{B}{T_c} \quad (3.4)$$

B is the bandwidth (Hz) and  $T_c$  is total chirp period including idle time (s).  $T_c$  can be reformulated further by knowing that the maximum number of collected ADC samples during a unit of time is defined by equation 3.5:

$$ADC\_samples = T_c \times Sampling\_Rate \quad (3.5)$$

If we substitute  $T_c$  in formula 3.5, the final form of range resolution can be obtained as :

$$Range_{Res} = \frac{c \times SamplingRate}{2 \times S \times ADCsamples} \quad (3.6)$$

$$Range_{Res} = \frac{c \times SamplingRate}{2000 \times Slope \times ADCsamples}$$

$$Range_{Res} = \frac{3 \times 10^8 (m/s) \times 6222 \times 10^6 Hz}{2000 \times \frac{42 \times 10^6 Hz}{10^{-6} s} \times 512}$$

$$Range_{Res} = 0.043 \text{ m} = 4.3 \text{ cm}$$

- **Repetition period for chirp 0 and chirp 1 :**

$$Rep_{Chirp} = Chirp\_Idle\_Time + Profile\_Idle\_Time + Ramp\_End\_Time \quad (3.7)$$

$$Rep_{Chirp_0} = 0 + 7 + 87.28 = 94.28 \mu s$$

$$Rep_{Chirp_1} = 0 + 7 + 87.28 = 94.28 \mu s$$

- **Speed resolution :**

Based on equation 2.10, it can be re-formulated and written in other form as follows:

$$Speed_{Res} = \frac{1000}{Rep_{Chirp_0} + Rep_{Chirp_1}} \times \frac{1}{ChirpNumber} \times \frac{\lambda(mm)}{2} \quad (3.8)$$

From memory address (index) of each chirp, total number of chirps in a given frame is given by :

$$\text{Chirp\_Number} = (\text{End\_Index} - \text{Start\_Index} + 1) \times \text{Loop\_Number}$$

$$\text{Number}_{\text{Chirp}_0} = (128 - 128 + 1) \times 32 = 32$$

$$\text{Number}_{\text{Chirp}_1} = (129 - 129 + 1) \times 32 = 32$$

Current wavelength in mm is :

$$\lambda = \frac{\text{Speed\_Of\_Light}}{\text{Start\_Frequency}} = \frac{3 \times 10^8 (\text{m/s})}{77 \times 10^9 \text{Hz}}$$

$$\lambda = 3.9 \times 10^{-3} \text{m} = 3.9 \text{ mm}$$

Finally, if we substitute total number of chirps and current wavelength in formula 3.8, we obtain :

$$\text{Speed}_{\text{ResChirp}_0} = \frac{1000}{94.28 \mu\text{s} + 94.28 \mu\text{s}} \times \frac{1}{32} \times \frac{3.9 \text{ mm}}{2}$$

$$\text{Speed}_{\text{ResChirp}_0} = 0.32 \text{ m/s} = 1.15 \text{ Km/h}$$

$$\text{Speed}_{\text{ResChirp}_1} = 0.32 \text{ m/s} = 1.15 \text{ Km/h}$$

- **Maximum unambiguous speed :**

As previously in speed resolution, maximum speed can be re-formulated, based on equation 2.9, one has :

$$\text{Speed}_{\text{Max}} = \frac{\lambda}{4T_c} = \frac{\lambda \times N}{2 \times T_c \times N} \times \frac{1}{2}$$

$$\text{Speed}_{\text{Max}} = \frac{\text{Speed}_{\text{Res}} \times N}{2}$$

The new form of maximum speed is :

$$\text{Speed}_{\text{Max}} = \frac{\text{Speed}_{\text{Res}} \times \text{ChirpNumber}}{2} \quad (3.9)$$

Finally maximum unambiguous speed for each chirp can be calculated as follows:

$$\text{Speed}_{\text{MaxChirp}_0} = \frac{\text{Speed}_{\text{ResChirp}_0} \times \text{Number}_{\text{Chirp}_0}}{2}$$

$$\text{Speed}_{\text{MaxChirp}_0} = \frac{0.32 \text{ m/s} \times 32}{2}$$

$$\text{Speed}_{\text{MaxChirp}_0} = 5.16 \text{ m/s} = 18.6 \text{ Km/h}$$

$$\text{Speed}_{\text{MaxChirp}_1} = 5.16 \text{ m/s} = 18.6 \text{ Km/h}$$

### 3.3.3 SRR-80m Profile

It is almost the same structure profile as the previous USRR-20m profile, but here the number of chirps is different where we define 128 chirps, the first 64 have an idle time of 3  $\mu$ s, and the remaining 64 have an idle time of 14.8  $\mu$ s (11.8  $\mu$ s extra idle time). Figure 3.10 shows the structure of this chirp within a given frame.

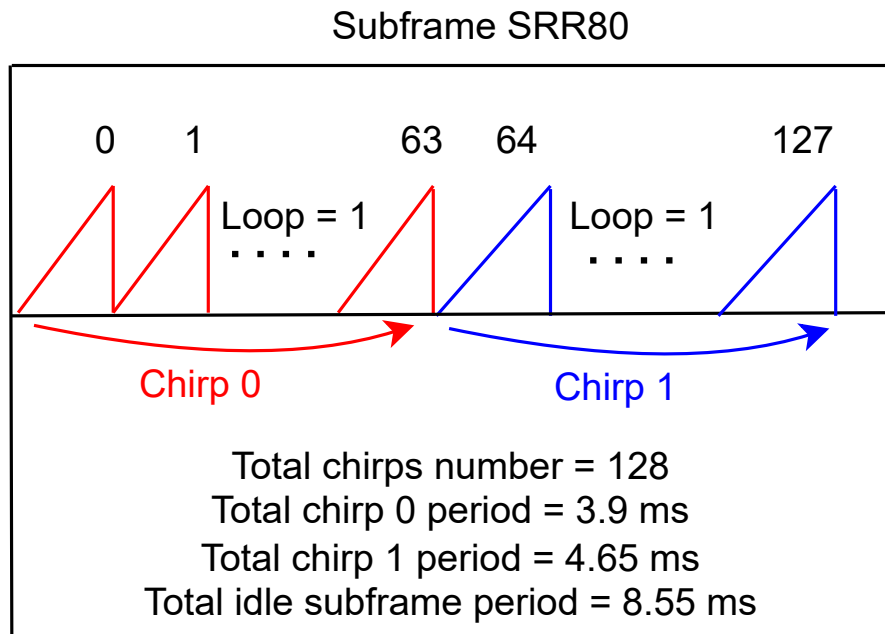


Figure 3.10: Single sub-frame for SRR-80m mode.

Tables 3.8(a), 3.8(b) and 3.9 show the corresponding three set of parameters of profile, chirp and frame respectively:

Profile ID	0
Sampling rate	5000 KSamp/s
Number of ADC samples	256
Start frequency	76 GHz
Frequency slope	8 MHz/s
Idle time	5 $\mu$ s
Ramp end time	56 $\mu$ s
TX start time	1 $\mu$ s
ADC start time	4.8 $\mu$ s

((a)) SRR-80m Profile parameters.

	Chirp 0	Chirp 1
Profile ID	0	0
Start index	0	64
End index	63	127
Start frequency variation	0	0
Frequency slope variation	0	0
Idle time variation	0	11.8
ADC start time variation	0	0
TX antenna	1	1

((b)) SRR-80m Chirp parameters.

Table 3.8: SRR-80m profile and chirp parameters.



Start index	0
End index	127
Loop count	1
Periodicity	30 ms
Number of real ADC samples	$256 \times 2 = 512$
Number of complex ADC samples	256
Chirp 0 number	$1 \times 64 = 64$
Chirp 1 number	$1 \times 64 = 64$
Total chirp number	$1 \times 128 = 128$
Transmitters number	1
Number of angle bins	32

Table 3.9: SRR-80m Sub-frame parameters.

### 3.3.4 Derived parameters - SRR-80m

- **Maximum detected range:**

Using equations 2.4 and 2.5 we have :

$$Range_{Max} = \frac{0.9 \times 5000 \times 10^6 (Samp/s) \times 3 \times 10^8 (m/s)}{2000 \times \frac{8 \times 10^6 Hz}{10^{-6}s}}$$

$$Range_{Max} = 84 \text{ m}$$

- **Range resolution:**

Based on equation 3.6, we have :

$$Range_{Res} = \frac{c \times SamplingRate}{2000 \times Slope \times ADCsamples}$$

$$Range_{Res} = \frac{3 \times 10^8 (m/s) \times 5000 \times 10^6 Hz}{2000 \times \frac{8 \times 10^6 Hz}{10^{-6}s} \times 256}$$

$$Range_{Res} = 0.36 \text{ m} = 36 \text{ cm}$$

- **Repetition period for chirp 0 and chirp 1 :**

Chirp 1 is a bit longer than chirp 0 since it has an additional idle time which results a different value for speed parameters for each chirp, based on equation 3.7, one has :

$$Rep_{Chirp_0} = 0 + 5 + 56 = 61 \text{ } \mu s$$

$$Rep_{Chirp_1} = 11.8 + 5 + 56 = 72.8 \text{ } \mu s$$

- **Speed resolution:**

After calculating repetition period of each chirp and number of total chirps in the frame, one substitutes values in equation 3.8:

$$Number_{Chirp_0} = (63 - 0 + 1) \times 1 = 64$$

$$Number_{Chirp_1} = (127 - 64 + 1) \times 1 = 64$$

$$Speed_{ResChirp0} = \frac{1000}{61 \mu s} \times \frac{1}{64} \times \frac{3.9 \text{ mm}}{2}$$

$$Speed_{ResChirp0} = 0.49 \text{ m/s} = 1.76 \text{ Km/h}$$

$$Speed_{ResChirp1} = 0.41 \text{ m/s} = 1.47 \text{ Km/h}$$

- **Maximum speed:**

One can substitute previous speed resolution and total chirp number in equation 3.9, one has two results for each chirp and since they have different values of repetition period they have different maximum speeds. Chinese remainder theorem was used to give the real estimation of speed in this profile which was about 90 Km/h.

$$Speed_{MaxChirp0} = \frac{0.49 \text{ m/s} \times 64}{2}$$

$$Speed_{MaxChirp0} = 15.7 \text{ m/s} = 56.5 \text{ Km/h}$$

$$Speed_{MaxChirp1} = 13.1 \text{ m/s} = 47.1 \text{ Km/h}$$



# Chapter 4

## Profile design

### 4.1 Profile structure

A profile is the basic timing template that is used to define the transmitted signal associated with all parameters that control the chirp in the frame. A frame would then consist of a sequence of chirps from a start index to an end index in the chirp configuration RAM which can be looped over up to 255 times. A single profile can be programmed to have multiple signals transmitted at once such that each signal can cover a specific range. But in this mode, attention to the trade-off aspect should be paid since it is not possible to have an LRR signal and low range resolution simultaneously. There are a set of controllable parameters per profile that allow us to supervise the entire performance of the system. Each chirp in the chirp configuration RAM can have small dither values that are added to the profile parameters defined in the profile RAM. Figure 4.1 shows the memory allocation of both chirp and profile.

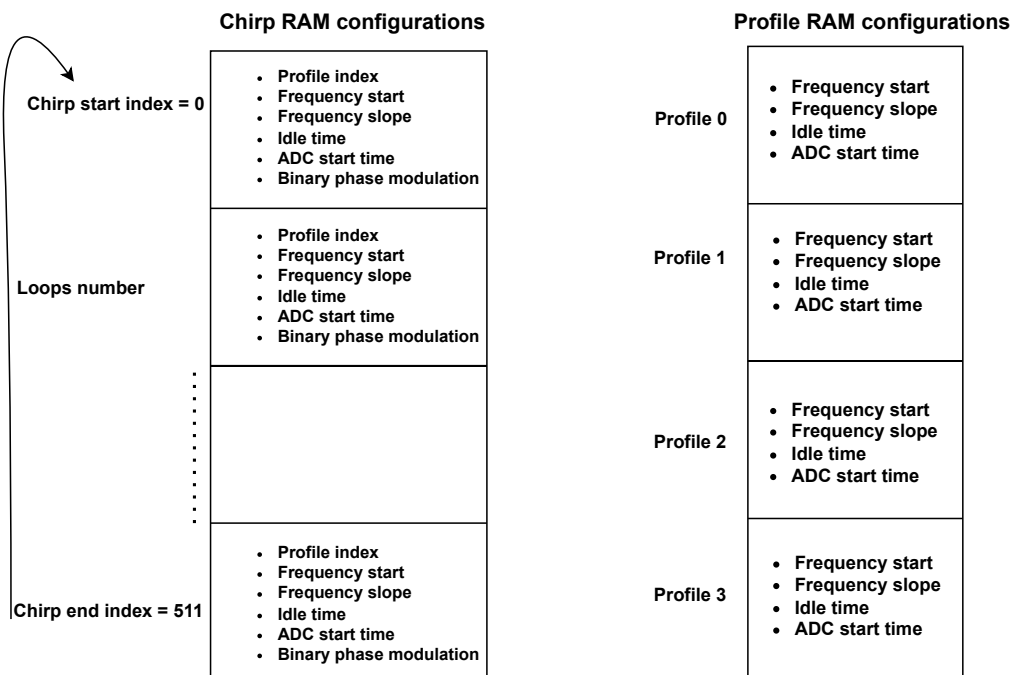


Figure 4.1: Chirp and profile allocation memory.

### 4.1.1 Timing parameters limitations:

In chapter 3, timing parameters have been introduced and their impacts on the system in general, we can now introduce limitation values for them that can not be exceeded during the programming procedure as we introduce an important timing parameter '**ADC Sampling Time**', which was not defined explicitly in the documentation. However, it was extracted due to many errors have been encountered during the configuration of the device.

Parameter	description	threshold value
Idle time	impacts directly speed and indirectly range	[1-7] $\mu$ s
TX start time	impacts directly speed and indirectly range	1 $\mu$ s
ADC start time	impacts directly speed and indirectly range	[1-40] $\mu$ s
Ramp end time	impacts directly maximum speed	[20 $\mu$ s - 5 ms]
<b>ADC sampling time</b>	impacts directly range resolution	88% of ramp end time

Table 4.1: Timing parameters threshold values.

The last parameter in table 4.1 was not clearly declared in different documents that describe the device. However, it was extracted during configuring the device and after several failed tests, it can be defined as the useful part of the chirp where ADC samples are collected and impact directly the range resolution, its value was estimated via many tests which is equal to 88% of chirp period. Based on formula 3.5, if we have a chirp duration = 30  $\mu$ s and ADC sampling rate = 5 Mbps for instance, therefore ADC sampling time = 26.4  $\mu$ s. The maximum number of ADC samples that can be collected in each sampling period is:

$$Num\_ADC\_Samp = 5 \times 10^6 \times 26.4 \times 10^{-6}$$

$$Num\_ADC\_Samp = 132 \text{ Samples}$$

For a successful and proper device programming, it is quite important to not violate formula 3.5.

### 4.1.2 A valid profile example

In this section, a detailed explanation of a valid profile will be given, and how it can be designed. We start our design by defining a set of 30 commands, each of which has a set of configurable parameters, the entire list of commands corresponds to a given profile. The following profile is for the USRR-20m but another version of what was defined in chapter 3 in tables 3.6(a), 3.6(b) and 3.7, it basically supports higher maximum speed (15.2 m/s). Each command could be represented by a table of parameters and each with its corresponding value, only the most important commands will be presented as the following tables :

RX channel enable	TX channel enable	cascading
15	3	0

Table 4.2: Channel configuration.

Number of ADC bits	ADC output format
2	3

Table 4.3: ADC configuration.

Subfram IDx	ADC output format	sample swap	channel interleave	chirp threshold
-1	1	0	1	0

Table 4.4: ADC buffer configuration.

Profile ID	start frequency	idle time	ADC start time	Ramp end time	TX output power	TX phase shifter
0	77 GHz	2 $\mu$ s	4 $\mu$ s	30 $\mu$ s	0	0

Table 4.5: Profile configuration 1.

frequency slope	TX start time	number of ADC samples	digital output sampling rate	HPF1 corner frequency	HPF2 corner frequency	RX gain
30 MHz/ $\mu$ s	1 $\mu$ s	150	6250 Ks/s	0	0	30 dB

Table 4.6: Profile configuration 2.

	chirp start IDx	chirp end IDx	profile ID	freq var	freq slope var	idle time var	start time var	TX mask
Chirp 0	0	0	0	0	0	0	0	1
Chirp 1	1	1	0	0	0	0	0	1

Table 4.7: Chirp configuration.

Chirp start index	Chirp end index	number of loops	number of frames	frame periodicity	trigger select	frame trigger delay
0	1	68	0	100 ms	1	0

Table 4.8: Frame configuration.

Subfram IDx	feature enable	chirp 0 index	chirp 1 index
-1	0	0	1

Table 4.9: BPM configuration.

Subfram IDx	detected obj	log mag range	noise profile	range-azi hmap	range az-elv hmap	rge-Dopp hmap	status info
-1	1	0	1	0	0	0	1

Table 4.10: GUI configuration.

	Subf ID	proc dir	op mode	noise win	guard length	cum noise	cyclic mode	thresh scale	peak group-ing
Range direction	-1	0	2	8	4	3	0	15 dB	1
Speed direction	-1	1	0	8	4	1	1	15 dB	1

Table 4.11: CFAR configuration.

### 4.1.3 Configuration commands

Previous tables 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 4.10 and 4.11 explain the detailed argument values of channel, ADC, ADC buffer, profile, chirp, frame, BPM, GUI, and CFAR configuration commands respectively of USRR-20m profile.

Figure 4.2 depicts a snapshot of the configuration file (.cfg) that is used to program the device. The first 55 lines are basically comments to clarify the different characteristics of the corresponding profile. Programming commands start with 'sensorStop' command at line 56 which stops any operation that the sensor performs currently before uploading the new configurations. Figure 4.3 depicts the entire configuration commands of USRR20 mode.

Table 4.12 explains what each command is used for, along with arguments associated with each one. However, it has been chosen the most important commands to explain inside this table. Argument values variation could have a big impact on system performance, or errors in some cases if values were not well calculated.

```

1  % Profile for Ultra Short Range Radar USRR(20m)
2  % Carrier frequency      GHz          77
3  % Ramp Slope            MHz/us       30
4  % Num ADC Samples      150
5  % ADC Sampling Rate Mps 6.25
6  % ADC Collection Time  us      0.88 x 30 = 26.4
7  % Extra ramp time required (start time) us 4
8  % Chirp time (end time - start time) us 26
9  % Chirp duration (end time) us      30
10 %
11 % Sweep BW (useful) MHz 900
12 % Total BW MHz         4000
13 % Max beat freq (80% of ADC sampling rate) MHz 5
14 % Max distance (80%) m 25
15 % Range resolution m   0.16
16 % Range resolution (meter per 1D-FFT bin) m/bin 0.16
17 %
18 % Inter-chirp duration us 2
19 % Number of chirp intervals in frame - 68
20 % Number of TX (TDM MIMO) 2
21 % Number of Tx elevation antennas 0
22 % Number of RX channels - 4
23 % Max unambiguous relative velocity kmph 54.4
24 % mileph 33.8
25 % Max extended relative velocity kmph 108.8
26 % mileph 67.6
27 %
28 % Frame time (total) ms 4.08 (30 us x 68 x 2 )- with idle time
29 % Frame time (active) ms 3.8 (28 us x 68 x 2 )- without idle time
30 % Range FFT size - 256
31 % Doppler FFT size - 64
32 % Radar data memory required KB 256
33 % Velocity resolution m/s 0.23
34 % Velocity resolution (m/s per 2D-FFT bin) m/s/bin 0.23
35 % Velocity Maximum m/s 15.12
36 % Extended Maximum Velocity m/s 30.24
37 % Maximum sweep accorss range bins range bin -
38 %
39 % By modifying num of salpmes, range resolution was getting better and better
40 %
41 % Another important thing is the relationship between num of samp, samp
42 % rate and ADC sampling time which equals 88% of ramp end time, the fact here
43 % is to make attention of Max num of samp that we can take and not exceed it !
44 %
45 % In frame config. commnad 68 for chirp 0 and 68 for chirp 1
46 % So in total 136 chirps/frame , we see that in Visualizer output
47 %
48 % Sweep BW (useful) can be computed by ( S=B/Tc),B = 30 MHz/us x 30 us = 900 MHz
49 %
50 % Max distance computed using its formula where IF(max) = 0.8 x Samp rate !
51 %
52 % The key word is when we choose small Tc to ensure higher Max speed
53 % slope should be high so shorter Max range , and B is high as well
54 % so larger Range Res! ==> Trade-off between Speed and Range
55 %
56 sensorStop
57 flushCfg
58 dfcDataOutputMode 1

```

Figure 4.2: Snapshot of profile USRR20 .cfg file.

```
59 channelCfg 15 3 0
60 adcCfg 2 1
61 adcbufCfg -1 0 1 1 1
62 %
63 profileCfg 0 77 2 4 30 0 0 30 1 150 6250 0 0 30
64 %
65 chirpCfg 0 0 0 0 0 0 0 1
66 chirpCfg 1 1 0 0 0 0 0 1
67 %
68 bpmCfg -1 0 0 1
69 %
70 frameCfg 0 1 68 0 100 1 0
71 %
72 lowPower 0 1
73 guiMonitor -1 1 0 0 0 0 0
74 cfarCfg -1 0 2 8 4 3 0 15.0 1
75 cfarCfg -1 1 0 8 4 4 1 15.0 1
76 %
77 multiObjBeamForming -1 0 0.5
78 calibDcRangeSig -1 0 -5 8 256
79 %
80 extendedMaxVelocity -1 1
81 clutterRemoval -1 0
82 %
83 compRangeBiasAndRxChanPhase 0.0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0
84 measureRangeBiasAndRxChanPhase 0 1.5 0.2
85 %
86 aoaFovCfg -1 -90 90 -90 90
87 cfarFovCfg -1 0 0 20.0
88 cfarFovCfg -1 1 -8.3 8.3
89 %
90 CQRxSatMonitor 0 3 5 127 0
91 CQSigImgMonitor 0 127 4
92 analogMonitor 0 0
93 lvdsStreamCfg -1 0 0 0
94 calibData 0 0 0
95 sensorStart
```

Figure 4.3: Snapshot of profile USRR20 .cfg file



Configuration command	description	arguments
dfeDataOutputMode	defines in which mode radar is operated.	"1" for frame based chirps and "3" for advanced frame conf.
channelCfg	how many desired antennas to be enabled.	as table 4.2 shows, "15" and "3" enable 4 RX and 2 TX respectively, "0" for cascading.
adcCfg	defines ADC desired operation mode.	as table 4.3 shows, "2" means that num. of ADC bits = 16 and "1" that complex 1X mode.
adcbufCfg	defines ADC buffer conf.	as table 4.4 shows, "-1" for legacy mode,"0" for complex mode,"0","0" by default and "1" was recommended for AWR1642 EVM.
profileCfg	most important API command where we define timing parameters and other parameters.	as tables 4.5 and 4.6 show, idle time, ADC start time, ramp end time, freq slope and num. of ADC samp are where our region to change the state of system. The remain arguments are by default or according to the documentation.
chirpCfg	defines the shape of chirp we seek and corresponding TX.	as table 4.7 shows, "0","1", chirp IDx, "0" means to which profile this chirp belongs to,"0","0","0","0" by default and "1" for transmission TX.
frameCfg	defines the shape of frame and number of chirps loops we seek.	as table 4.8 shows, "0","1", chirp IDx, "68" means to num. of each chirp, "0" means transmission of infinite num. of frames, "100 ms" every 100 ms create a frame, "1" and "0" by default.
bpmCfg	BPM API conf. in MIMO radar devices to improve SNR by running 2 TX simultaneously.	as table 4.9 shows, "-1" on which frame we want apply BPM,"0" disable BPM, "0" and "1" are chirp IDx.
guiMonitor	defines plot config. message to datapath.	as table 4.10 shows, "-1" for legacy mode,"1" enable detection of range or Doppler XY plan, the remain arguments, "1" for enabling and "0" for disabling.
cfarCfg	determine the power threshold for reflection process.	as table 4.11 shows, "-1" for legacy mode,"0" and "1" enable CFAR for range and Doppler detection, the remain arguments except last one were inspired by source code, "1" to detect only high reflector objects.
clutterRemoval	defines detection of only dynamic objects.	as figure 4.3 shows, "-1" for legacy mode, "1" for enabling the algorithm.
extendedMaxVelocity	a simple technique for velocity disambiguation to correct target velocities up to $2V_{Max}$ .	as figure 4.3 shows, "-1" for legacy mode,"1" for enabling the algorithm.

Table 4.12: configuration file .cfg explanation.

#### 4.1.4 Followed methodology

1. Figure 4.3 shows the full configuration process, the entire programming process contains 30 different commands. At the very first, we start by determining in which mode we like the device to operate single-mode or an advanced one. Then we initialize both antennas and ADC.

2. **'channelCfg:'**

In the channel configuration command, the first argument to set the 4 available RX antennas where  $0x1111b = 15$ , the second argument for enabling both azimuth antennas using bitmask  $0x11b = 3$ , and the third argument cascading we set it to  $0x0000$  which means single mmWave sensor application where this is the only choice available for AWR1642 device.

3. • **'adcCfg:'**

In the first argument we set the number of desired bits, only one option is available which is  $2 = 16$  bits, second argument denotes in which mode ADC is operated, real or complex mode, this field can take values of 0(Real), 1(Complex 1X), 2(Complex with Image band 2X). In the complex 2X mode, both imaginary and real IF spectrums are filtered and sent to ADC, thus, if the sampling rate is X, ADC data would include a frequency spectrum from  $-X/2$  to  $X/2$ . Whereas in complex 1X mode, only the real IF spectrum is filtered and sent to ADC, i.e. If the sampling rate is X, ADC data would include a frequency spectrum from 0 to X. Based on a given sampling rate, complex 1X provides twice the useful IF frequency and eventually Max Range. But the image band contains noise and interference signals which do not exist in complex 1X.

• **'adcbufCfg:'**

The first argument represents the sub-frame index, this field must be equal to "1" as long as the device is operated in single mode, but for advanced frame mode, it should be set to either the intended sub-frame number or "-1" to apply the same configuration to the all sub-frames. The second field is set to "0" indicating that the output of ADC is in complex mode, the second two fields are sample swap and channel interleave set to "1" by default, and the last field is chirp threshold can take values [0-8], it was recommended to set it to "1" for device AWR1642.

4. **'profileCfg:'**

• Then profile API, denotes how we really need our signal to be, it contains coarse parameters (14) of FMCW. Fine dithering values need to be programmed in chirp configuration, some of these parameters should be in match w.r.t. other APIs, otherwise the device outputs an error.

• One starts by indicating profile identifier, it varies within [0-3], start frequency for each profile, AWR1642 device is 77GHz (77 GHz - 81 GHz), then idle time can be set to any value less than  $< 7 \mu s$ , next ADC start time is the time of starting of ADC capture relative to the knee of the ramp, it can be set to any value  $< 40 \mu s$ , however, the smaller is the larger useful signal. Next, ramp end time (chirp duration) can be tuned to any value  $< 5 ms$ , this field is an important one for maximum speed. The next two parameters are set to '0' by default. Then, the frequency slope denotes how fast the signal is within a time interval, the maximum value is  $100 MHz/\mu s$ , TX start time must be always  $= 1 \mu s$ .

• Then, the number of ADC samples that can be collected in one sampling cycle, its maximum value is directly related to sampling rate and ADC sampling time based

on equation 3.5, the higher is the better range resolution. It can vary within [2-1024] samples in complex mode.

- Next parameter is sampling rate, any value can be chosen while ensuring it is < 6.25 MHz, maximum detected range is directly proportional to sampling rate in complex 1X mode according to equations 2.4 and 2.5.
- The next two parameters are for both HPF corner frequencies where they are set to 175 KHz and 350 KHz respectively. The last field represents RX antenna gain, any even value within [24-52] dB can be chosen.

#### 5. 'chirpCfg:'

Chirp configuration API includes the number of chirps in a frame, the selection of corresponding TX and BPM for a chirp. The first and second parameters correspond to start and end indices, the valid range is [0-511] and these defined chirps need to be included in the frame configuration structure to create an FMCW frame. The third parameter must be in match the profile identifier in the profile API. The last four parameters are 0 by default.

#### 6. 'frameCfg:'

In frame API, chirp sequence is defined, first and second arguments denote start and end indices respectively, start index varies within [0-511], and end index varies within [start index-511]. The third argument denotes the number of desired loops such that number of times to repeat from chirp start index to chirp start index in each frame, valid range is [1-255], this field value is inversely proportional to speed resolution. Then, the number of frames to transmit, valid range is [0-65535] where 0 means infinite frames. Next, frame periodicity represents the frame repetition period such that :

$$\text{PERIOD} \geq \text{Sum\_total\_time\_of\_all\_chirps} + \text{InterFrameBlankTime} \quad (4.1)$$

where equation 4.2 must be respected to have a valid profile :

$$\text{Sum\_total\_time\_of\_all\_chirps} = \text{Num\_Loops} \times \text{Num\_chirps} \times \text{Chirp\_Period} \quad (4.2)$$

'InterFrameBlankTime' is primarily for sensor calibration/monitoring and it must be  $\geq 300 \mu\text{s}$  typically. Valid range for frame periodicity is [300  $\mu\text{s}$  -1.342 s]. Last two parameters are set to 1 and 0 respectively by default.

#### 7. 'bpmCfg:'

BPM API configuration in MIMO radar devices is a technique that can be used to improve SNR by running 2 TX antennas simultaneously. When using this scheme, we should enable both azimuth TX in the chirp API by setting the last parameter to 3 indicating that each transmitted chirp is sent from both 2 TX antennas. It contains four fields, the first one to indicate under which mode radar operates, the second one to enable BPM, third and fourth ones, are start indices for both defined chirps.

#### 8. 'guiMonitor:'

GUI API represents plot settings, sub-frame argument is set to -1 indicating that single-mode operation, detected object parameters can take values of 0,1 or 2, it should be set to 1 or 2 enabling detection of objects and Doppler in XY plane. The third and fourth parameters are set to 1 if we like to relative signal power of the transmitted signal and noise figure in real-time. The fifth, sixth and seventh parameters are for heat map plots, they are usually not enabled since they require a very high measurement rate. Status information is a plot to show statistics information (CPU load, margins, device temperature readings, etc).

### 9. 'cfarCfg'

CFAR API is an algorithm performed after range and speed FFT. The role of the Constant False Alarm Rate circuitry is to determine the power threshold above which any return can be considered to probably originate from a real target and not from spurious sources. Two commands should be written, one for range detection and the other for speed detection as table 4.11 shows. The third parameter is operation mode, for range detection is 2 where in speed direction is 0. The remaining parameters are set as was recommended by the datasheet and source code, the last parameter is peak grouping, with the peak-grouping scheme enabled, instead of reporting a a cluster of detected neighboring points, only one point, the highest one, is reported.

10. In this section, the most relevant and important configuration commands were explained, however, the remaining ones are necessary and mandatory for a proper programming process.

## 4.2 Programming the device using PuTTY

It is a terminal emulator used for making serial connection operations and transfer data network. It was downloaded and linked to AWR1642 EVM, then both COM ports (data and configuration) were defined, flushing of the device has been done to be put in functional mode, and finally, configuration process is ready to be performed. Figure 4.4 shows the device while it is running successfully for a given profile:



Figure 4.4: Successful device programming using PuTTY terminal emulator.

Entering commands should be done very carefully since many issues have been arisen during configuration, such as invalid parameters especially in '**profileCfg**' API due to violation of

equation 3.5 was made for instance. Figure 4.5 shows a snapshot of terminal while sensor was running. The left side is the configuration commands and right side is the output of device while capturing data.

Figure 4.5: Snapshot on PuTTY terminal emulator while device is running.

### 4.3 mmWave Sensing Estimator

In the programming process, some considerations need to be taken to determine the value for each of these parameters, to determine the actual numbers for each of these parameters, the "**mmWave Sensing Estimator**" was used which is a tool to help in deriving timing parameters. It enables rapid prototyping of the chirp through real-time feedback of the chirp configuration and out-of-bounds checking. mmWave Sensing Estimator offers a very easy environment for radar concepts and summarizes sophisticated radar equations into simple-to-use configurable inputs. However, it is not necessary to adopt the recommended values to configure the device wherein most of the cases failure results, and error messages during the programming were shown due to different reasons such as the number of ADC samples that are desired or out-of-bound device capability.

Figure 4.6 shows a incorrect configuration of an application profile up to 125 m. To solve the problem a few adjustments need to be done in order to get valid parameters. And, as mentioned previously if a valid and correct configuration were achieved by the estimator it is not necessary to be the case when applying it to the device and another consideration should be taken as well.

One starts configuration by choosing the device model and specifying the bandwidth desired 1 GHz or 4 GHz, the higher the bandwidth the better range resolution, transmit power can be set to the maximum value 22 dBm. AWR1642 device support 2 TX and 4 RX antennas and each

antenna provides a maximum gain of 48 dB. Then the most related parameters which are scene parameters, describe the scene that chirp was designed for. We have the flexibility such that trade-off is always presented, for example, the maximum detectable range limits the range resolution, and the same fact can be applied to speed parameters. Each scene parameter is directly related to the configuration parameter as follows :

- Maximum detectable range  $\iff$  sampling frequency and frequency slope (equation 2.4).
- Range resolution  $\iff$  total sweep bandwidth (equation 2.2).
- Maximum velocity  $\iff$  ramp slope and chirp end time (equation 2.9).
- Velocity resolution  $\iff$  chirp end time and number chirp Loops in a single frame (equation 2.10).
- Measurement rate  $\iff$  active frame time (equation 4.1).
- Typical detectable object  $\iff$  RCS which relies on the object characteristics (equation 3.1 and table 3.1).

Figure 4.7 depicts a valid configuration chirp after performing some adjustments.

The screenshot shows the 'mmWave Sensing Estimator' interface. On the left, there are three sections of configuration parameters:

- Regulatory Restrictions:** Frequency Range (GHz) is 77 - 81, Maximum Bandwidth (MHz) is 4000, and Transmit Power (dBm) is 22.
- Scene Parameters:** Ambient Temperature (deg Celcius) is 25, Maximum Detectable Range (m) is 125, Range Resolution (cm) is 40, Maximum Velocity (km/h) is 55, Velocity Resolution (km/h) is 2, Measurement Rate (Hz) is 10, and Typical Detectable Object (m<sup>2</sup>) is Car with a value of 5.
- Additional Parameters:** Detection Loss (dB) is 1, System Loss (dB) is 1, Implementation Margin (dB) is 2, and Detection SNR (dB) is 12.

At the bottom left are buttons for 'SAVE CONFIG', 'LOAD CONFIG', and 'RESET CONFIG'. On the right, there is a diagram showing 'Frequency Start', 'TX Start Time', and 'Transmitter is ON' duration. Below the diagram are two tables of parameters:

Chirp Configuration Parameters			
Frequency Start (GHz)	77	# of Chirp Loops	55
Frequency Slope (MHz/us)	15.02	Frame Periodicity (ms)	100
Frequency Slope Constant	311	Idle Time (us)	2
Sampling Rate (ksps)	13903	ADC Valid Start Time (us)	3.80
# of Samples per Chirp	347	Ramp End Time	29.76

Information Only Parameters			
Bandwidth (MHz)	446.83	# of Range FFT Bins	512
Beat Frequency (MHz)	12.51	# of Doppler FFT Bins	64
Chirp Cycle Time (us)	31.76	Range Interbin Resolution (cm)	27.11
Chirp Repetition Period (us)	64	Velocity Interbin Resolution (m/s)	0.48
Active Frame Time (ms)	3.52	Radar Cube Size (KB)	880

At the bottom right, there are three error messages:

Error 1: The radar cube size is larger than the available memory on the device  
 Tips: Increase the "Range Resolution"/"Velocity Resolution", decrease the "Maximum Range"/"Maximum Velocity", and/or reduce the "# of Rx Antennas"/"# of Tx Antennas"

Error 2: Sampling frequency is higher than the maximum sampling frequency (>6.25 Msps)  
 Tips: Decrease the "Maximum Velocity", decrease the "Maximum Detectable Range", and/or increase the "Range Resolution"

Error 3: Beat frequency is higher than the maximum I/F frequency (>5 MHz)  
 Tips: Decrease the "Maximum Velocity", decrease the "Maximum Detectable Range", and/or increase the "Range Resolution"

Figure 4.6: A non valid estimated profile parameters.

mmWave Sensing Estimator
Help

**Regulatory Restrictions**

Frequency Range (GHz)

Maximum Bandwidth (MHz)

Transmit Power (dBm)

**Scene Parameters**

Ambient Temperature (deg Celcius)

Maximum Detectable Range (m)

Range Resolution (cm)

Maximum Velocity (km/h)

Velocity Resolution (km/h)

Measurement Rate (Hz)

Typical Detectable Object (m<sup>2</sup>)

**Additional Parameters**

Detection Loss (dB)

System Loss (dB)

Implementation Margin (dB)

Detection SNR (dB)

**Chirp Configuration Parameters**

Frequency Start (GHz)	<input type="text" value="77"/>	# of Chirp Loops	<input type="text" value="31"/>
Frequency Slope (MHz/us)	<input type="text" value="4.92"/>	Frame Periodicity (ms)	<input type="text" value="100"/>
Frequency Slope Constant	<input type="text" value="102"/>	Idle Time (us)	<input type="text" value="2"/>
Sampling Rate (ksps)	<input type="text" value="4560"/>	ADC Valid Start Time (us)	<input type="text" value="4.70"/>
# of Samples per Chirp	<input type="text" value="230"/>	Ramp End Time	<input type="text" value="56.14"/>

**Information Only Parameters**

Bandwidth (MHz)	<input type="text" value="276.47"/>	# of Range FFT Bins	<input type="text" value="256"/>
Beat Frequency (MHz)	<input type="text" value="4.10"/>	# of Doppler FFT Bins	<input type="text" value="32"/>
Chirp Cycle Time (us)	<input type="text" value="58.14"/>	Range Interbin Resolution (cm)	<input type="text" value="53.91"/>
Chirp Repetition Period (us)	<input type="text" value="116"/>	Velocity Interbin Resolution (m/s)	<input type="text" value="0.54"/>
Active Frame Time (ms)	<input type="text" value="3.60"/>	Radar Cube Size (KB)	<input type="text" value="248"/>

No errors are found.

Figure 4.7: A valid estimated profile parameters.



# Chapter 5

## Experiments and Data analysis

In this chapter, some experiments will be presented along with discussing the results of each and investigate the output.

### 5.1 Drone detection

In this experiment, drone model DJI FPV 255 x 312 x 127 mm was tested in an open environment in the Science Park such that its flying altitude was alternating within [2-6] meters and its range w.r.t. the radar up to 12 m. The tested chirp signal was the USRR-20m profile since drone dimensions are not that large and consider to be a small object with a small RCS value  $\leq 0.5 \text{ m}^2$  according to table 3.1. Figure 5.1 shows the tested drone model and the chosen environment to fly the drone easily and securely.



Figure 5.1: Tested environment and DJI drone model.

Tests showed that AWR1642 device is not that efficient to be used to detect drones regardless of the used profile in the experiment and how good the range resolution is. Tests showed that the drone was visible to the radar up to 12 m in the azimuth range and 6 m in the elevation plane providing that at some points even if the drone was flying with an azimuth range  $> 12 \text{ m}$  or elevation range  $> 6 \text{ m}$  was visible. That could be explained by the continuous varying value of RCS of the drone at that point was high enough to reflect some energy back that allow the radar to mark it as a valid object. Figure 5.2 depicts a snapshot during the test where drone flies at range about 5 m in azimuth, about 3 m in elevation, and its speed about 1 m/s.



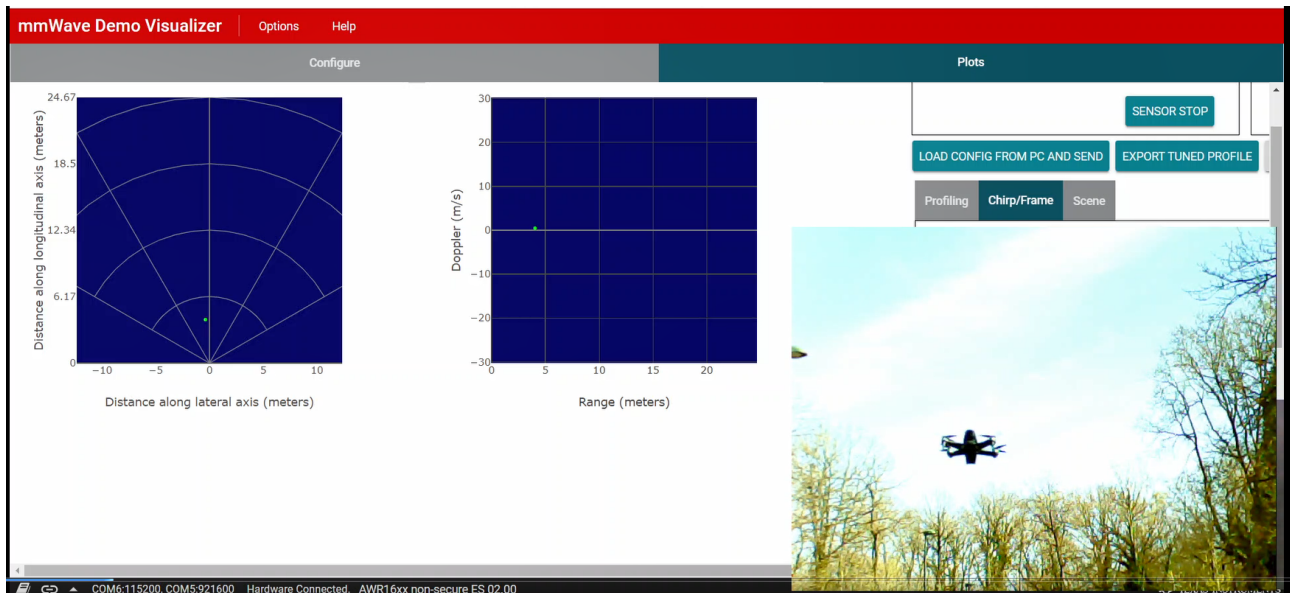


Figure 5.2: Snapshot of experiment output.

AWR1642 EVM provides  $\pm 20^\circ$  in elevation plane, as illustrated in figure 5.3 for a given ranges in both plans. However, it should be mentioned that in this test radar was not straight forward w.r.t. the azimuth plane and it was directed to the sky, however, the principle still valid and the same regardless position of the device.

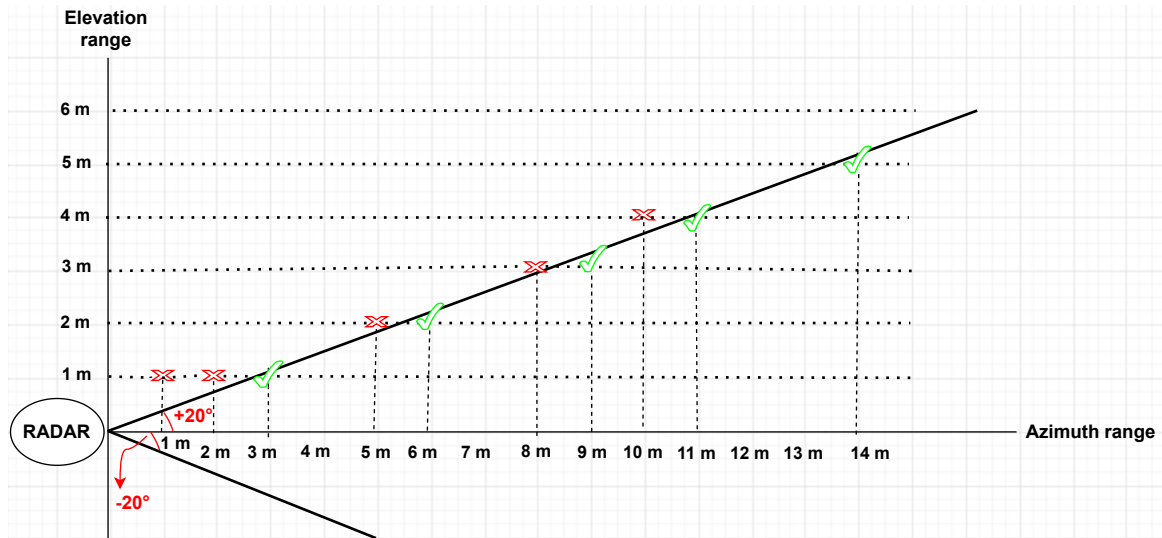


Figure 5.3: Visibility elevation FOV of AWR1642 EVM w.r.t. the drone.

Tables 5.1(a), 5.1(b), 5.2(a), 5.2(b) and 5.3(a) show when the drone was tested for different ranges and altitudes, the mark ✓ refers to 'detected' whereas the mark ✗ to 'not detected'. Previous tables show the drone at different ranges with different speeds while flying. Obtained results by tables confirm what was explained in figure 5.3 regarding the theoretical FOV in elevation. However, some cases were not matched and that could be explained by the variations of RCS value of the drone during the flight. As altitude increases, at some point, the drone is not visible anymore regardless of its altitude if it is larger than 5 m, in theory when altitude  $h = 6$  m, the drone should be detected starting from a range = 16.5 m but that is not the case in practice

Speed \ Range	0.5 m/s	1 m/s	1.5 m/s	2 m/s
1 m → 4 m	×	×	×	×
5 m	×	✓	✓	✓
6 m	✓	✓	✓	✓
7 m → 9 m	✓	✓	✓	×
10 m	✓	✓	×	×
11 m	✓	✓	×	×
12 m	✓	×	×	×
13 m	×	×	×	×

((a)) Drone detection at height 2 m.

Speed \ Range	0.5 m/s	1 m/s	1.5 m/s	2 m/s
1 m → 7 m	×	×	×	×
8 m	✓	✓	✓	✓
9 m	✓	✓	✓	×
10 m	✓	✓	✓	×
11 m	✓	✓	×	×
12 m	✓	×	×	×
13 m	×	×	×	×

((b)) Drone detection at height 3 m.

Table 5.1: Drone detection at height 2 m and 3 m.

Speed \ Range	0.5 m/s	1 m/s	1.5 m/s	2 m/s
1 m → 9 m	×	×	×	×
10 m	✓	✓	✓	×
11 m	✓	✓	×	×
12 m	✓	×	×	×
13 m	×	×	×	×

((a)) Drone detection at height 4 m.

Speed \ Range	0.5 m/s	1 m/s	1.5 m/s	2 m/s
1 m → 13 m	×	×	×	×
14 m	✓	✓	×	×
> 15 m	×	×	×	×

((b)) Drone detection at height 5 m.

Table 5.2: Drone detection at height 4 m and 5 m.

since less reflected energy arrives at RX and RCS value decreases which makes it hardly detected for high ranges and altitudes.

Table 5.3(b) illustrates the detection of the drone w.r.t. the azimuth angle for a given range up to 6 m. Angle values were estimated based on the alternating height of the drone within [2-6]m, from tests, it was observed that angle values decrease  $5^\circ$  when going deeply in the FOV by 1 m. These values don't necessarily be completely accurate and they were estimated based only on 2 experiments in Science Park. Thus, the higher range the tighter FOV angle in which drone can be seen until it seems to be totally perpendicular to the sensor at distance  $\approx 10$  m, after that range it is hardly seen by the sensor. That is normal since the higher distance between the detected object and sensor, the tighter FOV and the weaker the detection.

Figures 5.4, 5.5, 5.6, 5.7 and 5.8 depict binary plots of the detected points of the drone for different altitudes, these figures are the extension of tables 5.1(a), 5.1(b), 5.2(a), 5.2(b) and 5.3(a), in some cases, it might be noticed that there is a mis-match between the data in the binary plots in a hand and the data in the tables or in figure 5.3 in the other hand, that could be explained by the continuous changing of the rotation direction of the drone during the flight which changes the value of RCS of the drone and reflects higher energy when transmitted chirp hits the battery or any metal part on the surface of the drone for example. Based on the research that was published in IEEE in 2020 [2], the primary construction of the drone plays an important role in terms of drone detection, the research concluded that larger drones made of carbon fiber are easier to

Speed \ Range	0.5 m/s	1 m/s	1.5 m/s	2 m/s
1 m → 15 m	×	×	×	×
16 m	✓	✓	×	×
> 16 m	×	×	×	×

((a)) Drone detection at height 6 m.

Range (m)	Threshold angle value (Degrees)
1	up to 45° drone is ✓
2	up to 40° drone is ✓
3	up to 34° drone is ✓
4	up to 30° drone is ✓
5	up to 24° drone is ✓
6	up to 18° drone is ✓

((b)) Threshold angle values in azimuth plan w.r.t the radar.

Table 5.3: Drone detection at height 6 m and threshold angle values.

detect, whereas drones made from plastic (such as the drone in hand) and styrofoam materials are less visible to the radar systems. It has demonstrated that the large difference comes from the material nature of the tested drone such that the permittivity of the material significantly impacts the RCS, geometry and direction are also should be considered but they are quite hard to describe due to the complexity of the scattering mechanisms. Finally, the authors demonstrated by simulation that the battery position installed in the drone contributes to the RCS significantly and might be detected even if the drone is small and made of non-reflective materials, indeed, that was confirmed by this test since the tested drone model was hardly detected in most of cases.

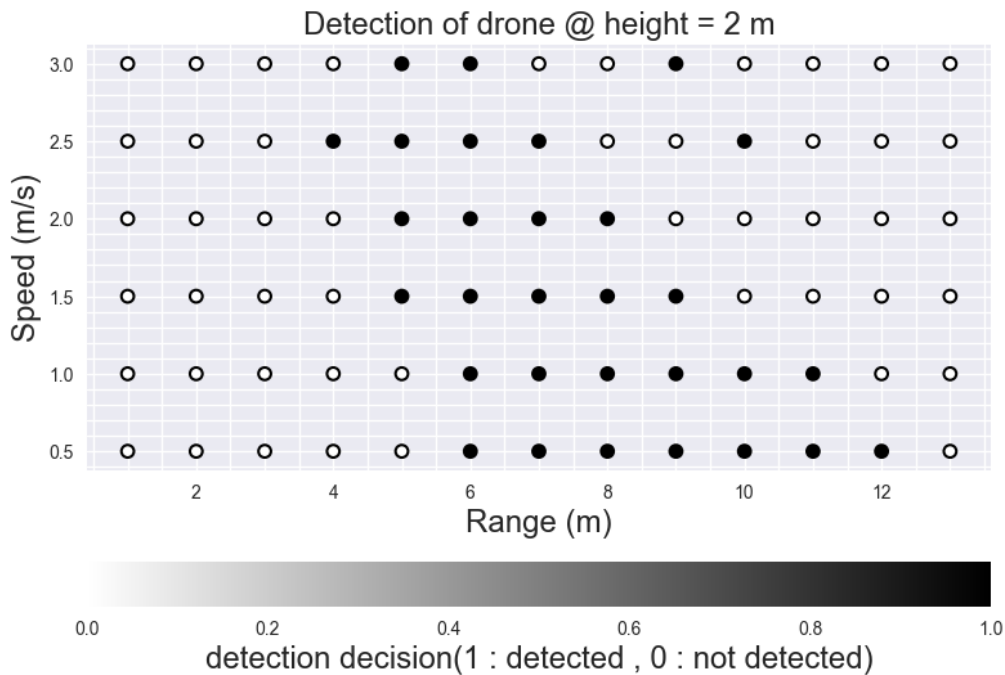


Figure 5.4: Detection at height of 2 m.

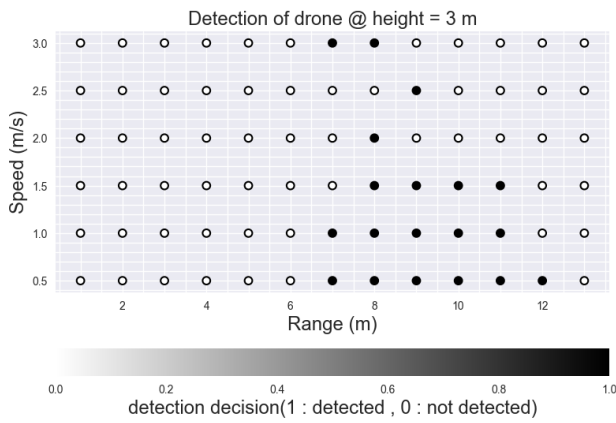


Figure 5.5: Detection at height of 3 m.

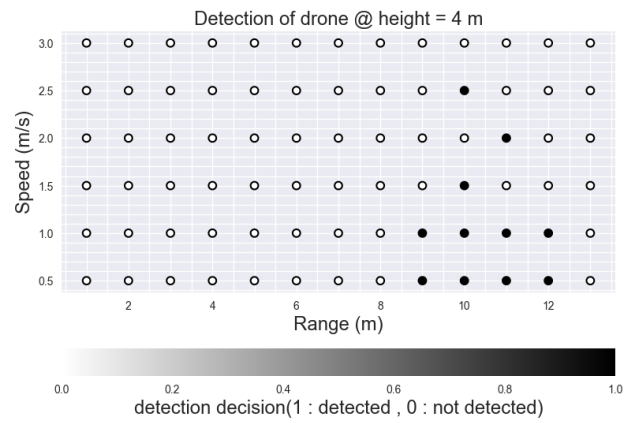


Figure 5.6: Detection at height of 4 m.

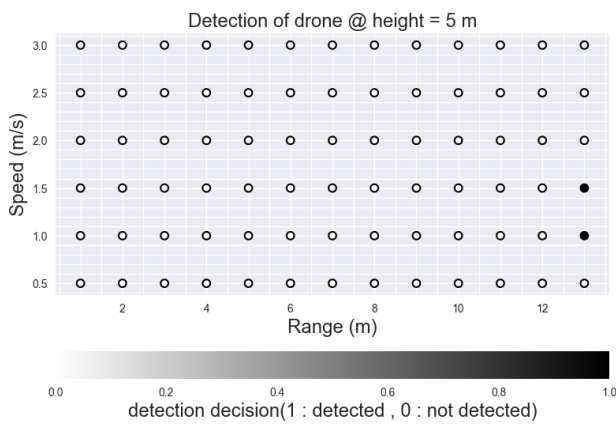


Figure 5.7: Detection at height of 5 m.

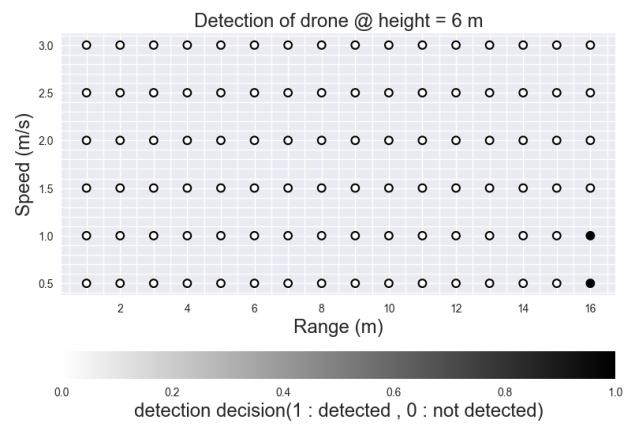


Figure 5.8: Detection at height of 6 m.

## 5.2 A car moving perpendicularly to the detection axis of the radar

This experiment was performed in the Science park for a car moving at different speeds such that its motion axis is perpendicular to the detection axis of the radar. Two locations have been tested, first location car moves at a fixed range w.r.t. the radar at 11 m as shown in the left side of figure 5.9 and in the second location, the car moves at a fixed range about 25 m as shown in the right side of figure 5.9. Location 1 has been chosen such that its FOV, in theory, can be covered entirely i.e  $140^\circ$  whereas location 2 has a smaller FOV due to trees that are presented in the right and left directions of the environment. Tables 5.4, 5.5 represent the detection values w.r.t. the AoA for both location respectively.

The profile that was used in this test has a maximum range of 80 m and a maximum speed of 37 Km/h (10.32 m/s). When the speed of the car is  $> 40$  Km/h measurement was not stable. It should be noticed that noise existed in both locations and sometimes was confusing to the external observer such that we read incorrect values for range and speed. One can see from the tables that detection is poor in azimuth compared to datasheet value  $70^\circ$  in both directions. For location 1, the FOV is less in the right direction. The best detection was when  $\Theta = 0^\circ$  in both

locations (i.e car is directly perpendicular to the sensor). In the location 2 test and from table 5.5, it could be noticed that FOV in the right direction is less than what is in 5.4 since there are more trees in location 2 that act as obstacles against transmitted and reflected signals. One can see that AoA is almost  $0^\circ$  in most of the cases. In both tests when the speed of the car is higher than what is configured by the radar, displayed speed values are not matched with actual ones, for example, the car moves at speed of 13.9 m/s away from the radar but the displayed speed value is -20 m/s (noisy measurement !). That could be explained due to less amount of reflected signals at RX from the moving car and more random scattered signals in the environment.



Figure 5.9: Car position in location 1 and location 2.

Speed Km/h (m/s)	direction w.r.t the radar in azimuth	threshold angle in azimuth	threshold detection angle
20 (5.5)	coming ←	$\Theta_{Left} \approx 50^\circ$ $\Theta_{Right} \approx 30^\circ$	it is ✓ as long as $\Theta \in [40^\circ - 50^\circ]$ in both directions.
	leaving →	$\Theta_{Right} \approx 30^\circ$ $\Theta_{Left} \approx 40^\circ$	
30 (8.3)	coming ←	$\Theta_{Left} \approx 40^\circ$ $\Theta_{Right} \approx 70^\circ$	it is ✓ as long as $\Theta < 70^\circ$ in both directions.
	leaving →	$\Theta_{Right} \approx 70^\circ$ $\Theta_{Left} \approx 40^\circ$	
40 (11.1)	coming ←	$\Theta_{Left} \approx 40^\circ$ $\Theta_{Right} \approx 30^\circ$	it is ✓ as long as $\Theta < 40^\circ$ in both directions.
	leaving →	$\Theta_{Right} \approx 30^\circ$ $\Theta_{Left} \approx 40^\circ$	
50 (13.9)	coming ←	$\Theta_{Left} \approx 30^\circ$ $\Theta_{Right} \approx 25^\circ$	it is ✓ as long as $\Theta < 30^\circ$ in both directions otherwise it is ✗.
	leaving →	$\Theta_{Right} \approx 30^\circ$ $\Theta_{Left} \approx 25^\circ$	

Table 5.4: Location 1, car perpendicular to detection axis.



Speed Km/h (m/s)	direction w.r.t the radar in azimuth	threshold angle in azimuth	threshold detection angle
10 (2.7)	going ←	$\Theta_{Right} \approx 0^\circ$ $\Theta_{Left} \approx 20^\circ$	it is ✓ as long as $\Theta \in [0^\circ - 20^\circ]$ in both directions.
	coming back →	$\Theta_{Left} \approx 20^\circ$ $\Theta_{Right} \approx 5^\circ$	
20 (5.5)	going ←	$\Theta_{Right} \approx 0^\circ$ $\Theta_{Left} \approx 20^\circ$	it is ✓ as long as $\Theta < 20^\circ$ in both directions.
	coming back →	$\Theta_{Left} \approx 0^\circ$ $\Theta_{Right} \approx 20^\circ$	
30 (8.3)	going ←	$\Theta_{Right} \approx 5^\circ$ $\Theta_{Left} \approx 20^\circ$	it is ✓ as long as $\Theta < 20^\circ$ in both directions.
	coming back →	$\Theta_{Left} \approx 0^\circ$ $\Theta_{Right} \approx 20^\circ$	
40 (11.1)	going ←	$\Theta_{Right} \approx 5^\circ$ $\Theta_{Left} \approx 20^\circ$	it is ✓ as long as $\Theta < 15^\circ$ in both directions.
	coming back →	$\Theta_{Left} \approx 20^\circ$ $\Theta_{Right} \approx 0^\circ$	
50 (13.9)	going ←	$\Theta_{Right} \approx 5^\circ$ $\Theta_{Left} \approx 15^\circ$	it is ✓ as long as $\Theta < 20^\circ$ in both directions otherwise it is ✗
	coming back →	$\Theta_{Left} \approx 20^\circ$ $\Theta_{Right} \approx 0^\circ$	

Table 5.5: Location 2, car perpendicular to detection axis.

Figures 5.10 and 5.11 depict the polar plots that correspond previous tables for location 1 and location 2 respectively.

According to the data shown in figure 5.10 it can be seen that we have approximate symmetry in detection w.r.t. the azimuth angles in different speeds except when the speed of the car is 30 Km/h, the radar was able to detect it in higher FOV ( $\approx 140^\circ$ ) than other speeds. However, it was not able to track the car continuously as it can be seen there was an invisible area in which radar lost the car  $[40^\circ - 70^\circ]$  and  $[125^\circ - 160^\circ]$ , which could be explained by the randomness of scattering phenomenon in which at that moment radar was not able to receive enough reflected energy such that its algorithm recognizes the car as a valid object. For speeds 20 Km/h and 30 Km/h, it can be observed that they have higher FOV than speeds 40 Km/h and 50 Km/h, in general, when speed = 20 Km/h the FOV  $\approx 120^\circ$  and when speed = 30 Km/h the FOV  $\approx 140^\circ$  in one hand, whereas when speed = 40 Km/h the FOV  $\approx 74^\circ$  and when speed = 50 Km/h the FOV  $\approx 60^\circ$  in the other hand, that is expected in fact since the tested profile provides detection up to 37 Km/h such that the higher speeds than programmed threshold one allowed by the radar, the less FOV. Again, due to the scattering effect when the object moves at high speeds it scatters more random signals than reflected ones to RX. Of course, when the speed of the car was 20 km/h and 30 Km/h, the device was able to detect and track it almost during the entire test, however, in this polar plot it was chosen only some points to be shown for illustration purposes but in fact, the motion of the car was seen as a circular pattern in the output of the device when the car moves at

higher speeds than allowed by the device this pattern becomes shorter. One last fact that could be noticed is the fact that indeed the detection is more optimal when the object is perpendicular to the detection axis of the device, in another word, the less AoA the better the detection as was demonstrated in chapter 1 by formula 2.13, that was confirmed by figure 5.10 where we can see more detected points next to azimuth angle =  $90^\circ$  or when AoA =  $0^\circ$ .

In figure 5.11 one can see the polar plot of the test at location 2. By default, the device provides FOV  $\approx 140^\circ$  however in this location there are trees and grasses on the left and right sides of the device with some more and denser on the right w.r.t. the left one, and since, the distance between the radar and the same car is higher than location 1 experiment, as a result, one expects less FOV and weaker detection. Moreover, as seen in the right side of figure 5.9 it is possible that these trees are preventing some amount of reflected signals to travel back to the RX of the device, which was confirmed by the polar plot where we have tighter FOV than what we had in location 1 test. As was explained in the previous section, indeed when speed is  $< 37$  Km/h, detection is better, we have more detected points and the tested object will be tracked and reported continuously since we still respect the speed condition of the profile, once the speed becomes higher, we have fewer detected points and less FOV consequently, furthermore, the object would not be tracked perfectly and the corresponding measurement will be associated with much noise.

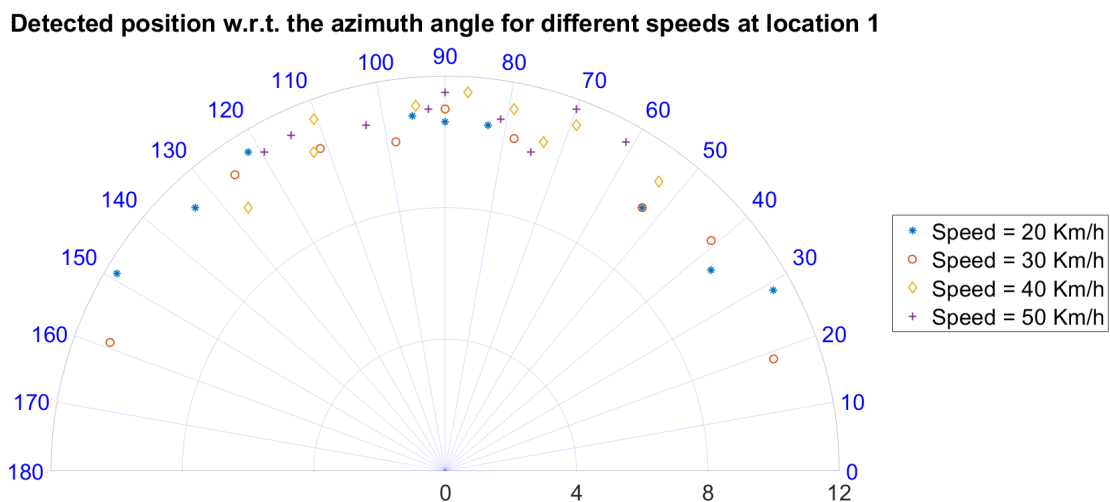


Figure 5.10: Detected pointed polar plot at location 1.

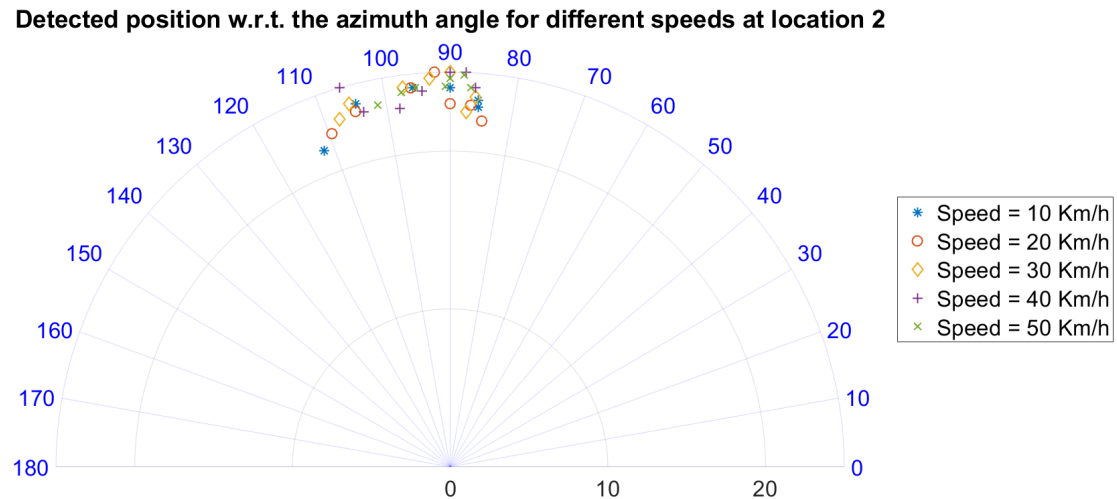


Figure 5.11: Detected pointed polar plot at location 2.

The module implements many algorithms that can be exploited to extract the desired information, such as range and noise profile information in real-time, the load on the CPU core, and heat maps. Figure 5.12 depicts two cars next to Montifiore institute (marked in the blue circle), at distance about 30 m from the device. Figure 5.13 depicts the corresponding range and noise profile for this scene, in fact, it shows the range profile at the 0th Doppler for tracked objects (static objects) using the blue line and noise profile (if enabled) using green line, it can be seen that there is a peak at that distance indicating the stationary car, and the green line is the noise that is presented in the environment. Other stationary cars are invisible since in this configuration we have disabled the detection of the static objects already by enabling **clutterRemoval** command to eliminate static clutter from the scene, we use this option only when trying to detect moving objects in the scene and static objects need to be masked out.

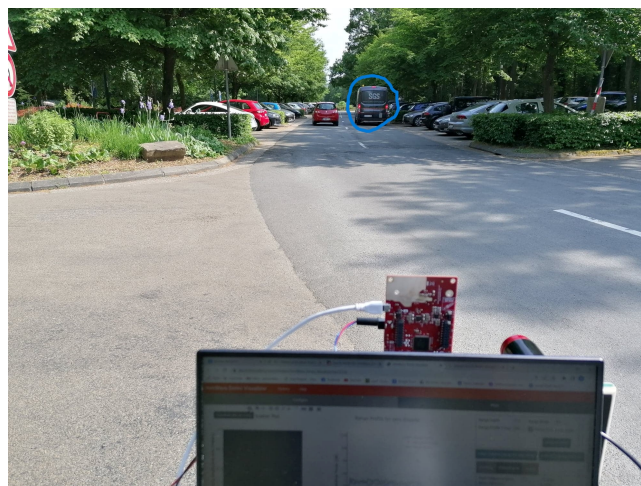


Figure 5.12: Two cars along longitudinal axis.



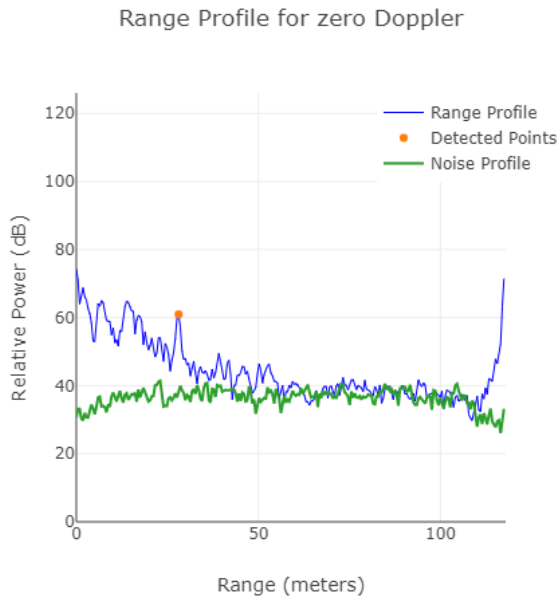


Figure 5.13: Range and noise profile.

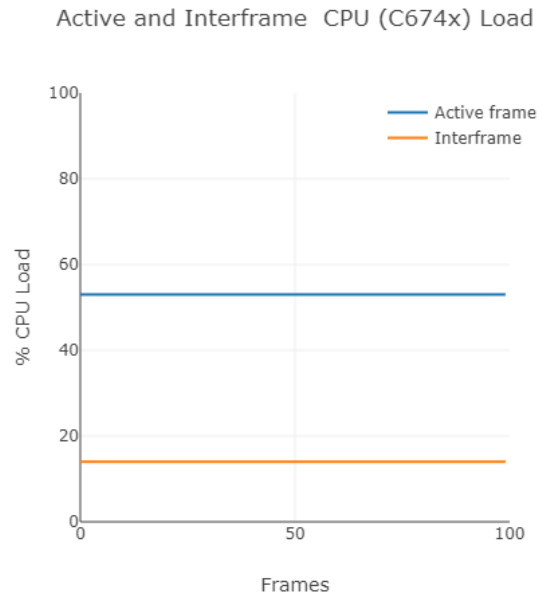


Figure 5.14: CPU load.

### 5.3 Metallic and non metallic pieces detection

They are a set of tests performed to discover the ability of the device detection of different materials and surfaces. During tests, it has been chosen several materials such as Aluminium, Copper, Cartoon, and Plastic of different sizes and thicknesses. Tests were performed in the CEDIA hall in one of the acoustic rooms which is a completely empty one and in the corridor next to CEDIA.

Figure 5.15 declares the maximum azimuth detected range for a given object in the vicinity of the radar w.r.t. the distance on the perpendicular axis. Four angle values have been chosen to make the tests  $0^\circ$ ,  $20^\circ$ ,  $40^\circ$  and  $55^\circ$ , yet, the device should provide  $70^\circ$  in both sides according to the datasheet but that is not the case. During these tests it was observed that maximum AoA in the FOV can be reached at  $60^\circ$  in both sides, according to the source code it was mentioned that to have the best possible angular resolution the radar was programmed by default to provide around  $60^\circ$  (based on formula 2.15 the smaller AoA, the better angular resolution). However, during the experiments it has been observed that AoA is not the only factor that impacts the visibility of a given object, there is another angle  $\Phi$  that must be considered which defines the angle between the object itself and the elevation axis such that there is a limitation of  $\Phi$  where if the object is tilted too much, it is not seen by the radar anymore regardless of its range w.r.t. to the radar.

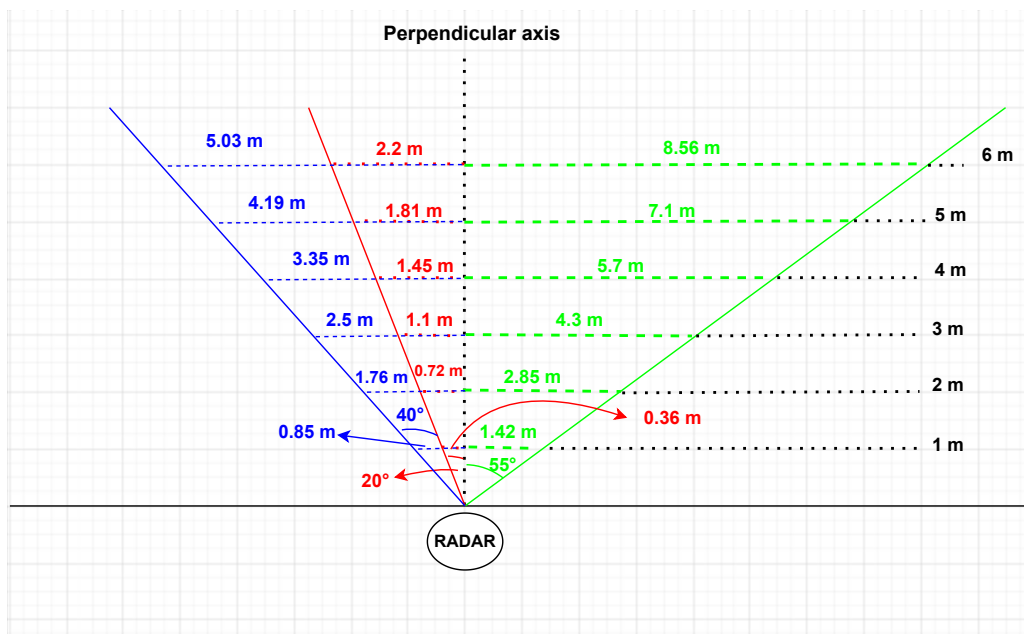


Figure 5.15: Maximum theoretical detected azimuth range for different angles.

The first test was on an Aluminum panel with dimensions 22 cm x 20 cm x 4 mm (length x width x thickness) in the corridor next to CEDIA, two angles to be taken into account based on the experiments when it comes to the object angular position w.r.t. the sensor and elevation axis.  $\Theta$  is the angle between sensor and object in the azimuth plane i.e AoA. Its maximum theoretical value is  $70^\circ$  but it will be considered up to  $55^\circ$  based on the recommendations.  $\Phi$  is the angle between object and elevation axis at object location, it is a critical angle that must be taken into account when it comes to detection procedure. As an example, figure 5.16 shows two metallic objects in the vicinity of the radar at a range no more than 1.5 m, yet none was seen by the radar because  $\Phi$  in this test was more than its allowed threshold value.

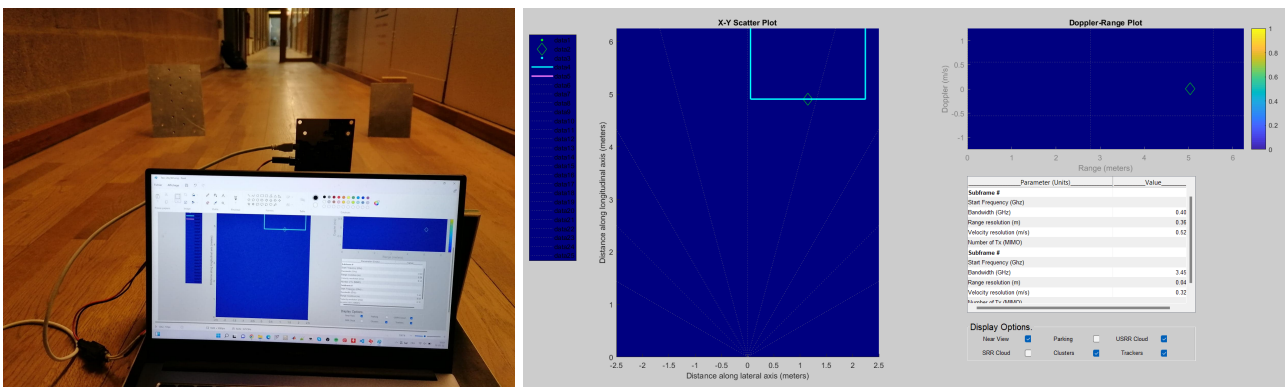


Figure 5.16: Non detected two metallic objects.

Tables 5.6 and 5.7 the threshold values of angle  $\Phi$  in which the metallic object is still visible to the sensor w.r.t. the four tested values of angle  $\Theta$ . It can be seen from tables that for a given object such as the tested one the detection is good as far as its distance w.r.t. the sensor does not exceed 5 m, otherwise, it is not visible unless  $\Theta = 0^\circ$ . As it can be observed that  $\Phi$  is inversely proportional to the range of an object such that at some point when distance  $\geq 10$  m, the object is not seen anymore regardless of the value of  $\Phi$ .

Range (m)	$\Theta$	$\Phi$	$\Theta$	$\Phi$
1	$0^\circ$	it is ✓ as long as $\Phi < 35^\circ$	$20^\circ$	it is ✓ as long as $\Phi < 30^\circ$
2	$0^\circ$	it is ✓ as long as $\Phi < 25^\circ$	$20^\circ$	it is ✓ as long as $\Phi < 23^\circ$
3	$0^\circ$	it is ✓ as long as $\Phi < 20^\circ$	$20^\circ$	it is ✓ as long as $\Phi < 19^\circ$
4	$0^\circ$	it is ✓ as long as $\Phi < 14^\circ$	$20^\circ$	it is ✓ as long as $\Phi < 10^\circ$
5	$0^\circ$	it is ✓ as long as $\Phi < 9^\circ$	$20^\circ$	it is ✓ as long as $\Phi < 6^\circ$
6	$0^\circ$	it is ✓ as long as $\Phi < 4^\circ$	$20^\circ$	it is ✓ as long as $\Phi < 2^\circ$ otherwise it is ✗
7	$0^\circ$	it is ✓ iff $\Phi = 0^\circ$	$20^\circ$	it is ✓ iff $\Phi = 0^\circ$
8	$0^\circ$	it is ✓ iff $\Phi = 0^\circ$	$20^\circ$	it is ✓ iff $\Phi = 0^\circ$
$\geq 10$	$0^\circ$	it is ✗	$20^\circ$	it is ✗

Table 5.6: Detection of Aluminum panel when angle  $\Theta$  is  $0^\circ$  and  $20^\circ$  respectively.

Range (m)	$\Theta$	$\Phi$	$\Theta$	$\Phi$
1	$40^\circ$	it is ✓ as long as $\Phi < 25^\circ$	$55^\circ$	it is ✓ as long as $\Phi < 20^\circ$
2	$40^\circ$	it is ✓ as long as $\Phi < 20^\circ$	$55^\circ$	it is ✓ as long as $\Phi < 16^\circ$
3	$40^\circ$	it is ✓ as long as $\Phi < 19^\circ$	$55^\circ$	it is ✓ as long as $\Phi < 14^\circ$
4	$40^\circ$	it is ✓ as long as $\Phi < 10^\circ$	$55^\circ$	it is ✓ as long as $\Phi < 6^\circ$
5	$40^\circ$	it is ✓ as long as $\Phi < 4^\circ$	$55^\circ$	it is ✓ as long as $\Phi < 2^\circ$
6	$40^\circ$	it is ✓ as long as $\Phi = 0^\circ$	$55^\circ$	it is ✓ as long as $\Phi = 0^\circ$ otherwise it is ✗
7	$40^\circ$	it is ✓ iff $\Phi = 0^\circ$	$55^\circ$	it is ✓ iff $\Phi = 0^\circ$
8	$40^\circ$	it is ✓ iff $\Phi = 0^\circ$	$55^\circ$	it is ✓ iff $\Phi = 0^\circ$
$\geq 10$	$40^\circ$	it is ✗	$55^\circ$	it is ✗

Table 5.7: Detection of Aluminum panel when angle  $\Theta$  is  $40^\circ$  and  $55^\circ$  respectively.

In figure 5.17 one can see the same metallic panels at some distance to the radar and some angle  $\Phi$ , in this scene we have two metallic panels at distance about 1.3 m to the device, then another metallic panel at distance about 2.8 m from the device but with different AoA w.r.t. the previous two panels, and, last object is basically a bag (marked in a red circle) at distance 3.6 m to the device, In this scene, all objects were visible to the device as we see as we still respect the condition of threshold value of angle  $\Phi$ . Pink zone is just to show the closest objects in the vicinity, and blue rectangle is the dBscan clustering algorithm to group objects with the same nature as much as possible. The bag was also detected and not clustered in the blue rectangle cluster indicating that this object (the bag) is different in nature than the three metallic panels. Left plot indicates the speed of detected objects (here is 0 m/s).

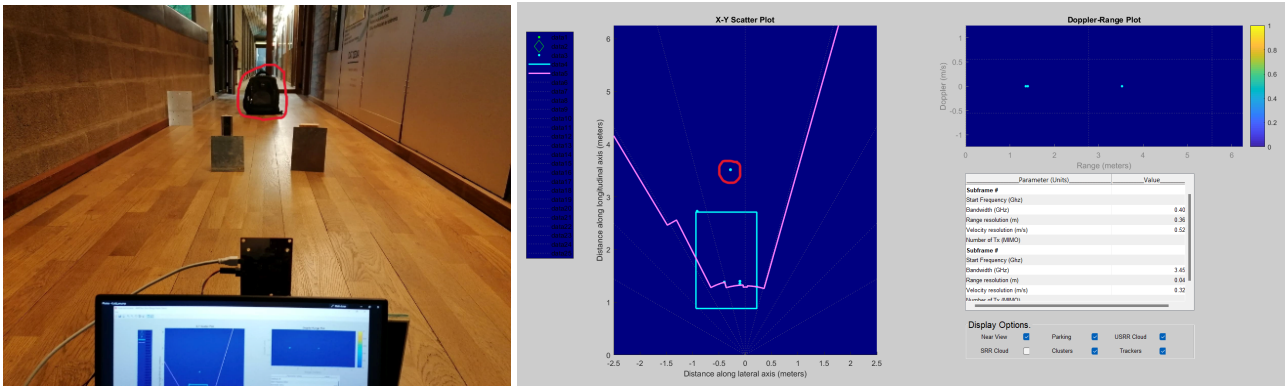


Figure 5.17: Four detected objects at different distances.

Figure 5.18 shows the echo room which is completely empty except the Plastic dust bin at distance  $\approx 3$  m where it can be clearly detected. In general RF signals can be reflected to all kind of conducting materials such that RX receives the reflected energy and display the detected object, whereas usually radar can not detect non-conducting materials and RF signals penetrating them. However, that is not always the case i.e radar can detect Cartoon or Plastic objects sometimes if they are too close or their surfaces are large enough to reflect some signals back to the radar as it can be seen in figure 5.19 in which dustbin is at distance  $\approx 4$  m and is not seen where cartoon box is. Clusters in figure 5.18 are possibly something in the concrete since they represent something out of the room boundaries.

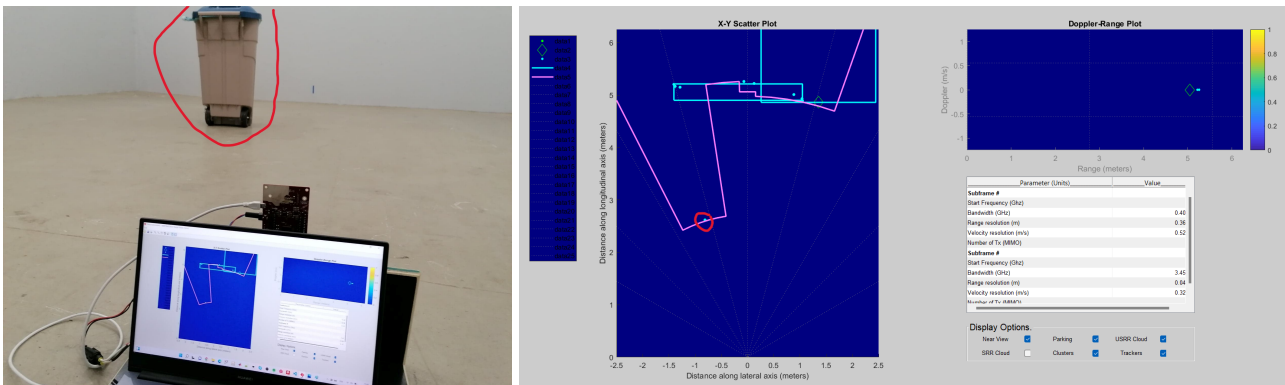


Figure 5.18: Detected Plastic dustbin at distance about 3m and AoA about  $20^\circ$ .

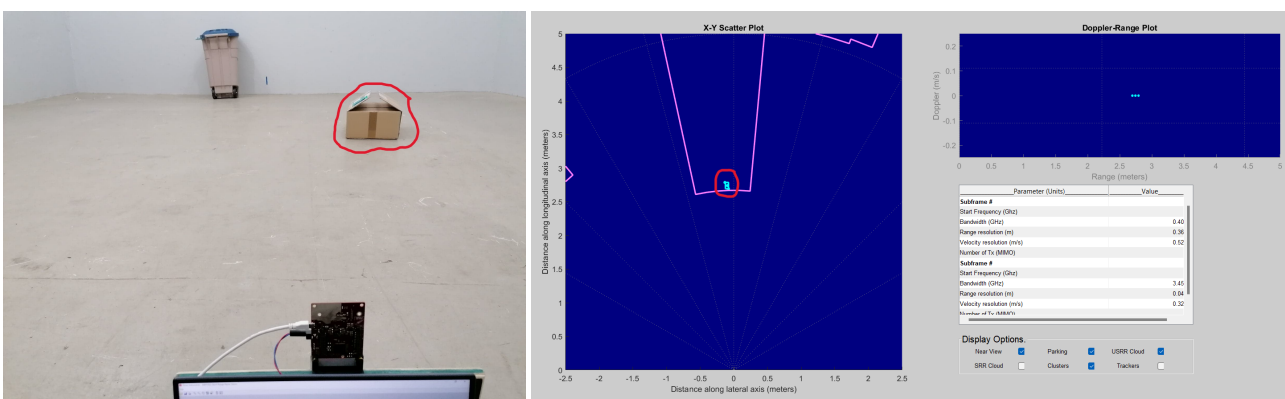


Figure 5.19: Non detected plastic dustbin and detected Cartoon box .

## 5.4 Building detection

This test aims basically to check the ability of the module to detect a building as a block itself. As light can not penetrate through solid walls, radio waves also get blocked by concrete, and whatever gets through is then blocked again on the way back. That is what was exactly confirmed by this test except when the test was performed against a building covered with metal pieces.

It was impossible to detect a building as a valid object if it does not have metallic surfaces regardless of its volume and its azimuthal angle w.r.t. the radar. However, when there are some metallic parts on the surface of the building, it is recognized as a valid object, when the azimuthal angle (AoA) between radar and the building changes, the detection was impacted, at some point there was a threshold such that any value higher than it, the building is not visible anymore. The best signal can be obtained is when the azimuthal angle =  $0^\circ$ .

From table 5.8 it can be seen that the threshold azimuthal angle is  $45^\circ$ , and from table 5.9 it can be seen that the building is not visible at all regardless angle value. Figure 5.20 depicts both tested buildings, on the left side musée en plein air du Sart-Tilman building (non-detected) and on the right side B52 building (detected), both are next to Montifiore institute.

Figure 5.21 shows a snapshot of the experiment that has been done against a part of a building next to B52, where at this point the distance between the radar and the building  $\approx 15$  m, it can be seen it is a completely invisible scene whatever distance and azimuthal angle are (it was forgotten to record the experiment of musée en plein air building), whereas in figure 5.22 where the radar is 10 m away from B52, it can be seen that we had a complete visible detection of B52 with azimuthal angle  $\approx 7^\circ$  w.r.t. the radar. In this test, profile USRR-20m has been used.

In figures 5.21 and 5.22, X-Y scatter plot shows basically the detected objects in the X-Y dimensions. When we have advanced frame configuration, this plot shows the union of points detected in all sub-frames of a given frame. Doppler\_Range plot in the right side of the output shows the speed of all detected objects in range-Doppler coordinates (here is 0 m/s), the same condition here as X-Y Scatter Plot when we have advanced frame configuration.

Angle between radar and building	$0^\circ$	$15^\circ$	$30^\circ$	$45^\circ$	$60^\circ$	$75^\circ$	$90^\circ$
Detection	✓	✓	✓	✓	✗	✗	✗

Table 5.8: Detection of a building covered with metallic material (B52 building)

Angle between radar and building	$0^\circ$	$15^\circ$	$30^\circ$	$45^\circ$	$60^\circ$	$75^\circ$	$90^\circ$
Detection	✗	✗	✗	✗	✗	✗	✗

Table 5.9: Detection of a building not covered with metallic material (Musée en plein air).





Figure 5.20: Musée en plein air and B52 buildings.

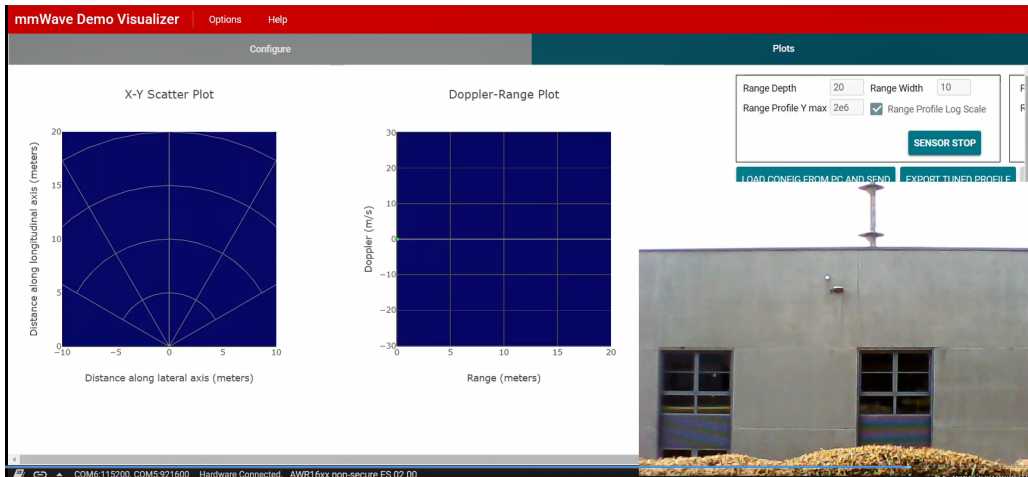


Figure 5.21: Invisible detection of a part of the building.

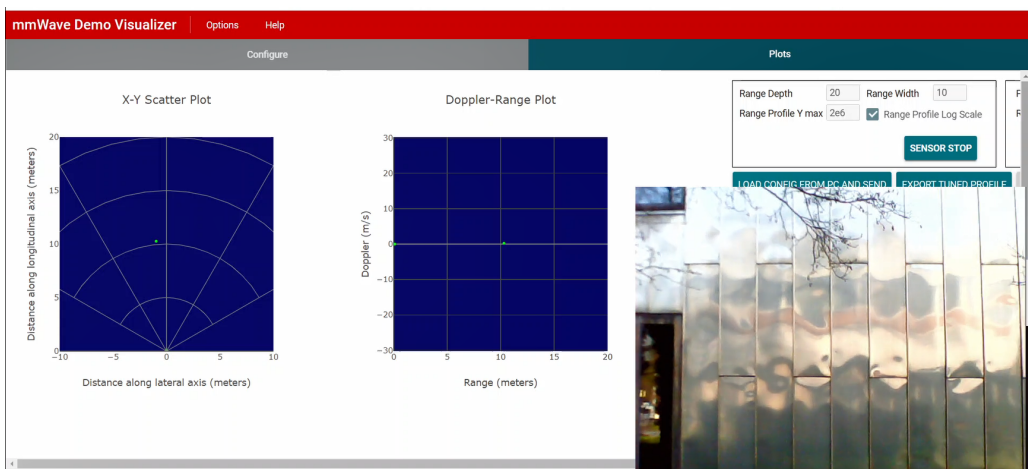


Figure 5.22: Visible detection of the B52 building.

## 5.5 A car moving in parallel to the detection axis of the radar

In this test, a car with different speeds was moving along the detection axis of the radar. The test aims to check the validity of the developed profile in terms of maximum range and speed that profile allowed. The profile that was used provides maximum detectable range of 125 m, maximum unambiguous speed of 48 Km/h (13.3 m/s) and range resolution of 0.8 m.

The experiment confirmed the efficiency of the developed profile where radar was able to track the car and reporting correct speed value in the real time, as it was less affected by the noise in the environment. Tested environment is shown in figure 5.23 as it provides long range to be checked.



Figure 5.23: The tested location 3 in the Science Park.

Table 5.10 illustrates the obtained results for three different speeds. Figures 5.24, 5.25 and 5.26 depict snapshots of the visualized output while car was moving at 50 Km/h, 30 Km/h and 70 Km/h respectively.

Actual speed	Direction w.r.t. the radar in azimuth	Detection	Speed reading
30 Km/h (8.3 m/s)	go ↑ and back ↓ directions	it is ✓ up to 92 m	≈8.3 m/s
50 Km/h (13.9 m/s)	go ↑ and back ↓ directions	it is ✓ up to 113 m	≈ -13.3 m/s
70 Km/h (19.4 m/s)	go ↑ and back ↓ directions	it is ✓ up to 80 m	≈ - 8.5 m/s

Table 5.10: Location 3, car parallel to detection axis of the radar.

When looking at figure 5.24, it can be seen that the radar was able to track the car up to 113 m which is quite interesting since the car was moving at a speed is a bit higher than maximum allowed speed by the profile. However, the reported speed value is -13.3 m/s which is not matched with the actual one, that could be explained due to actual speed > maximum allowed speed such that there is a speed margin of (13.9 m/s - 13.3 m/s = 0.6 m/s). In such a case, the displayed detected speed point in the Doppler plot is shifted to the negative part of the plot indicating that the



actual speed of the tracked object should be smaller to be displayed as a correct result. Or, there is another option which is enabling Extended Maximum Velocity API as **extendedMaxVelocity -1 1** as was explained in figure 4.3 and table 4.12 in chapter 4, such that the speed limit of axis would be in this case  $-26.6 \text{ m/s} \rightarrow +26.6 \text{ m/s}$ , however, it was noticed that when enabling this API, at high speeds it was displayed many objects which do not exist in the real time as noise and that is why it was decided not to enable it. In this test, we can observe that there is a person (blue circle) walking in the environment, however, he was not visible to the radar where his distance is  $> 20 \text{ m}$ , since radar can see small objects whose RCS  $< 1 \text{ m}^2$  up to that 20 m only and it needs low range resolution profile.

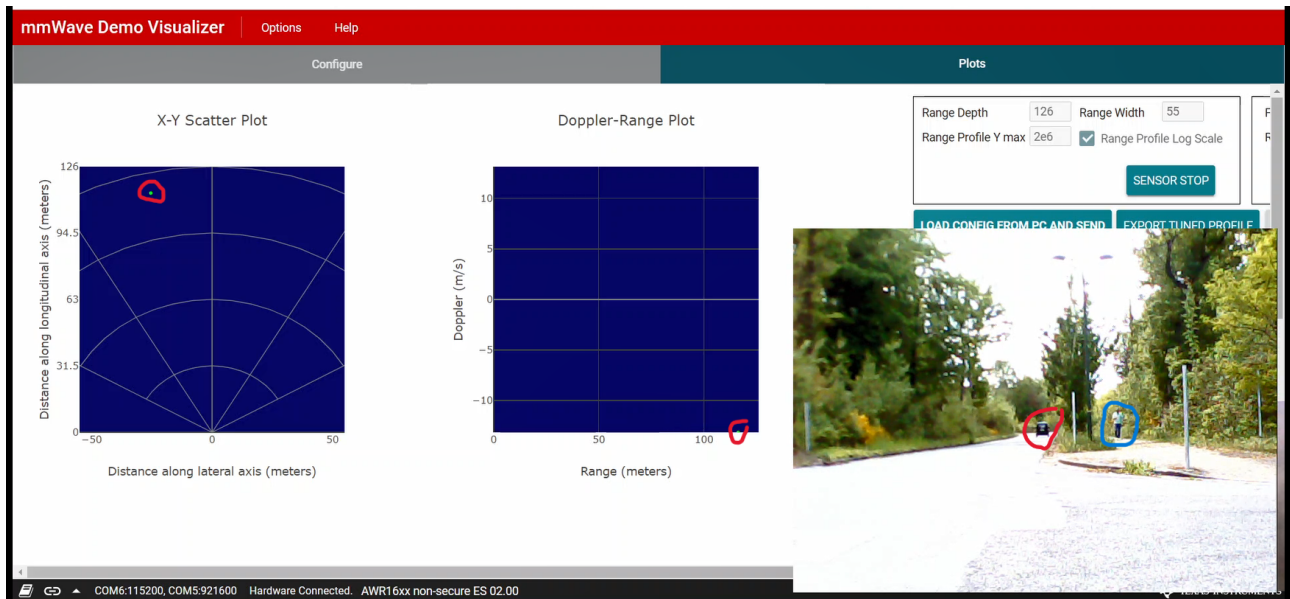


Figure 5.24: Location3, moving car at 50 Km/h.

Based on the reported output result in figure 5.25, it is clearly shown that the car has been tracked up to 92 m with lower AoA than what we have when the car was moving at a speed of 50 Km/h according to figure 5.24, lower AoA is simply because speed now is not  $>$  maximum allowed speed (48 Km/h), therefore, the car reflects higher amount of energy back to RX and less random scattered signals in the environment and that is why the car was shown in X-Y scatter plot as it is the real time (nearly perpendicular in azimuth  $\Theta \approx 5^\circ$ ). The reported speed in the Doppler plot ( $\approx +8.3 \text{ m/s}$ ) is correct and as was expected since the speed of the car did respect the profile condition.

The last case that has been examined is when the speed of the car was 70 Km/h as figure 5.26 depicts, since the speed condition has been violated, the car has been tracked up to 80 m lower than previous two cases), we have high AoA ( $\Theta \approx 15^\circ$ ) as we had when the car was moving at 50 km/h, however, this AoA does not really represent the actual AoA in real time of the tracked car due to speed condition in both cases. As was explained in the case of speed of 50 km/h, it can be seen that the detected point in the Doppler plot is presented in the negative part of the plot ( $\approx -8.5 \text{ m/s}$ ), speed margin here ( $19.4 \text{ m/s} - 13.3 \text{ m/s} = 6.1 \text{ m/s}$ ), this margin should be presented as a detected point in the negative part and that is what can be seen  $\approx 4.8 \text{ m/s}$ . The actual speed margin does not match with theoretical ones, which could be explained that possibly the speed of the car is not completely accurate at 70 Km/s, it is actually ( $13.3 \text{ m/s} + 4.8 \text{ m/s} = 18.1 \text{ m/s} = 65.16 \text{ km/h}$ ). Two detected points in blue are basically me when I was observing the crossroad to avoid



any potential accident.

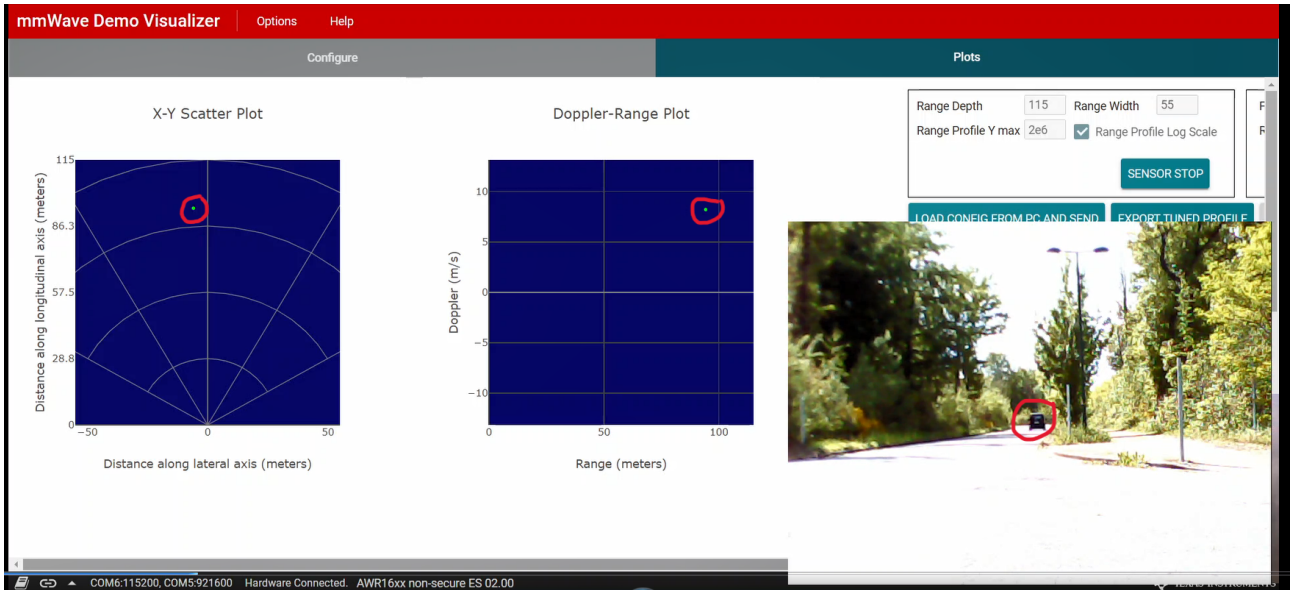


Figure 5.25: Location3, moving car at 30 Km/h.

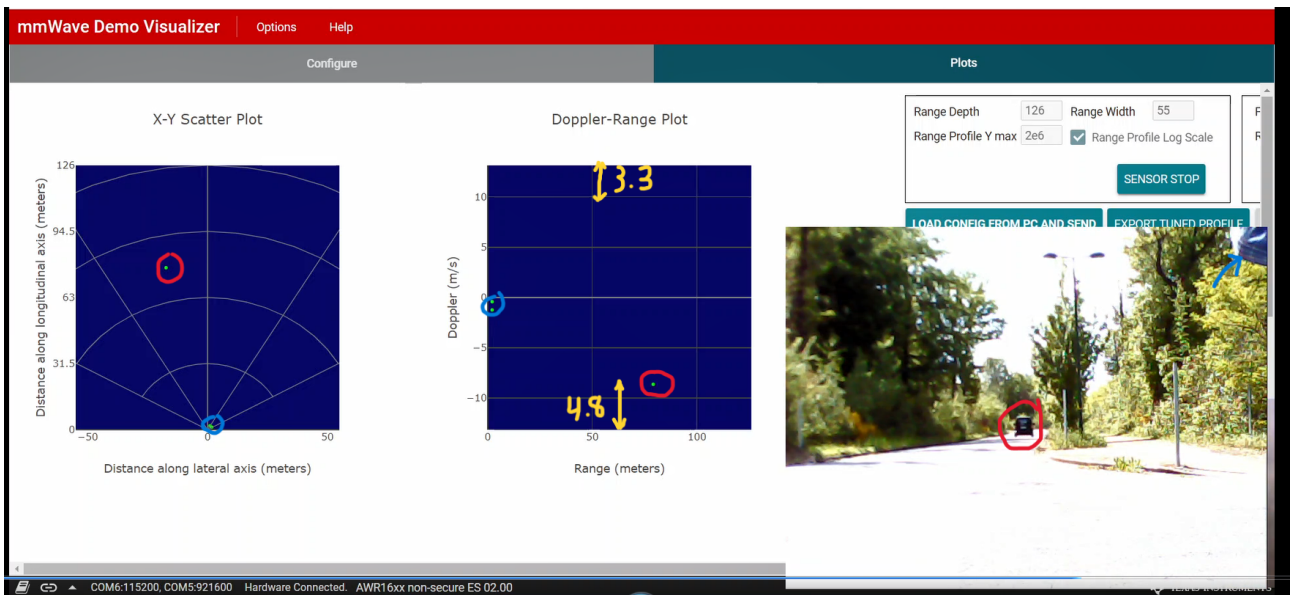


Figure 5.26: Location3, moving car at 70 Km/h.

# Chapter 6

## Conclusion and future work

In this research, the FMCW AWR1642 radar EVM from TI was studied and verified to check its ability for drone detection. Initially and based on some requirements which were needed, this EVM was chosen among several ones since it provides many benefits over others, such as improved performance, wider RF bandwidth of 77–81 GHz, highly linear chirps and faster ramps up to  $100\text{ MHz}/\mu\text{s}$ , which in turns, can be interpreted to better detection solutions such as maximum range and speed that can be provided by the EVM, without forgetting that it has low power consumption and cost.

When it has been used for the first time against some dynamic objects such as cars in an open environment, it was able to detect and track the moving cars up to 68 m at maximum for a given speed of about 56 Km/h. Due to that, it was essential to see how the device is operated and what kind of software process it follows to perform the detection, therefore, in chapter 3 a complete analytical study of the transmitted signal (chirp) was provided in order to be able to program the device and provide higher detected range and speed, it was such a critical mission to grasp the relationships between the FMCW chirp configuration and system performance parameters helps in selecting the right chirp configurations, and for that, it was the best way to do that is to understand figure 3.1 since it holds so much information needed for the thesis!.

After interpreting the source code in chapter 3 and understanding the structure of the frame, thus, it was possible in chapter 4 to develop our own signal and implement it in a given profile which was able to cover higher ranges and speeds in automotive applications since TI products allow considerable flexibility in designing the TX signal and multiple chirp types and various kinds of frame sequences to be programmed upfront. A complete development methodology was provided and explained step by step to reach a valid profile. However, it should be noticed that during profile design, some trade-off has been taken into account to make a valid signaling profile where we had to sacrifice some parameters at some phases such as range resolution for example since we were interested in detecting and tracking a given object as high range and speed as possible.

In chapter 5 several experiments have been performed using different signaling profiles according to the characteristics of each test. In the test of detection of metal and non-metal objects, it was clear that it works as expected for both types of objects but up to a low range, normally  $< 10\text{ m}$  for metal ones, and, up to 6 m for non-metal ones, as we have to take the size of each object into account. When performing the car experiment in both cases when it was moving perpendicularly to the detection axis at locations 1 and 2 and when it was moving in parallel to the detection axis of the device at location 3, the device proved it was as effective as possible when obtaining results in the field such that it detected and tracked the moving car up to 113 m and up to a speed of 70 Km/h. It was an improved result compared to the default performance. Furthermore, the profile

that has been used at location 3 test was LRR-125m, we can observe that the profile worked as expected and covered  $\approx 90\%$  of the theoretical maximum range (125m), and the allowed maximum speed by the profile was  $\approx 48$  km/h, however, the device tracked the moving car at 70 Km/h and that was due to the disambiguation process that works based on 'Chinese remainder theorem', (e.g if the car moves at speed 100 Km/h, it will be still detected and tracked but up to lower range).

Drone detection experiment was not so successful as previous ones using the AWR1642 EVM radar, tests showed that the tested drone model DJI fpv was not visible to the device in several places in the tested domain as desired, yet, the profile that has been used in this test USRR-20m provides low range resolution which is good for detection of the small objects. As stated before, the low value of RCS of the drone could be the reason that was not visible very well in many cases in terms of flying speed and altitudes. This device can not be used to detect drones in general unless the tested drone has enough metal materials on its surface and has a larger size than the DJI fpv model such that RCS value is high enough to be visible at higher speeds and altitudes. The device proved that it is very recommended and efficient to be used in automotive applications against cars and humans (at a lower range) if it is programmed with proper profiles.

### **Future work:**

In the current work that has been done during the thesis and based on what was stated above in the conclusion, one can try to make more profile development to cover higher ranges and speeds allowed by the configured profile. One can make more experiments as well for extending the measurements to expand the dataset and obtain new limitations for other materials, objects or even perform tests against a larger variety of drones with higher values of RCS using different frequency bands as well (possibly drone made of Carbon-Fiber material as was recommended). During the thesis, some research has been done in terms of the signal processing part of the device, but unfortunately, the work had not been continued due to lack of time. However, as was seen, the device implements many algorithms that work altogether to track the detected object and output the best estimation value of the range and the speed in XY plane, after the validation that this is a real object and not a false one using CFAR algorithm, it has been understood that the main algorithm that is used to estimate the range and speed is Extended Kalman Filter EKF, which is based on a system of dynamic state models to represent the range and the speed in two phases, predict and update, it performs some iterations to reach out the best estimation finally. One could continue the research by going deep into the source code and try to enhance the detection process and possibly reduce noise and interference signals on the reflected signal to increase the SNR.

# **Appendices**



# Appendix A

## Hardware setup

To be able to test the device efficiently and properly, a small wooden setup was made in CEDIA hall in Montifiore. A small battery was used to power the device with a proper voltage where it outputs 16 V, battery output then is connected to the input of the DC-DC converter to have an output voltage of 5 V that goes directly to power the device. Figure A.1 shows the complete setup.

All components were fixed and attached properly to the setup and the device was inserted inside a wooden sandwich which was designed and printed in the micro-sys lab as figure A.2 shows. This sandwich has the freedom to move left and right on the top edge of the setup to cover the specified FOV within a given environment. Next to the place of the module, a small camera was placed to record the test session simultaneously with the radar during the experiment so that we have a recorded video at the same time with the output of detected objects to be able to make a comparison between both from an external observer point of view. However, there is a tiny amount of delay time (few ms) between both radar output and camera output, so it should be taken into account, however, it has no big impact on recorded data. Figure A.3 shows the used camera.

An adapter connector was used to power the EVM, it was connected to the output of the converter by two wires such that there is no need to solder wires. Male and female connectors were used as well, from the battery to the converter to facilitate the charging of the battery each time is needed as figure A.5 shows. The battery that was used is a 4 cells Kokam LiPo battery, voltage of each cell  $\approx 4$  V to have a total output voltage of 16 V, it can be charged safely with current [2-3] A. Battery was fixed from the bottom side with the wooden setup with a non-conducting material for safety purposes as shown in figure A.4.

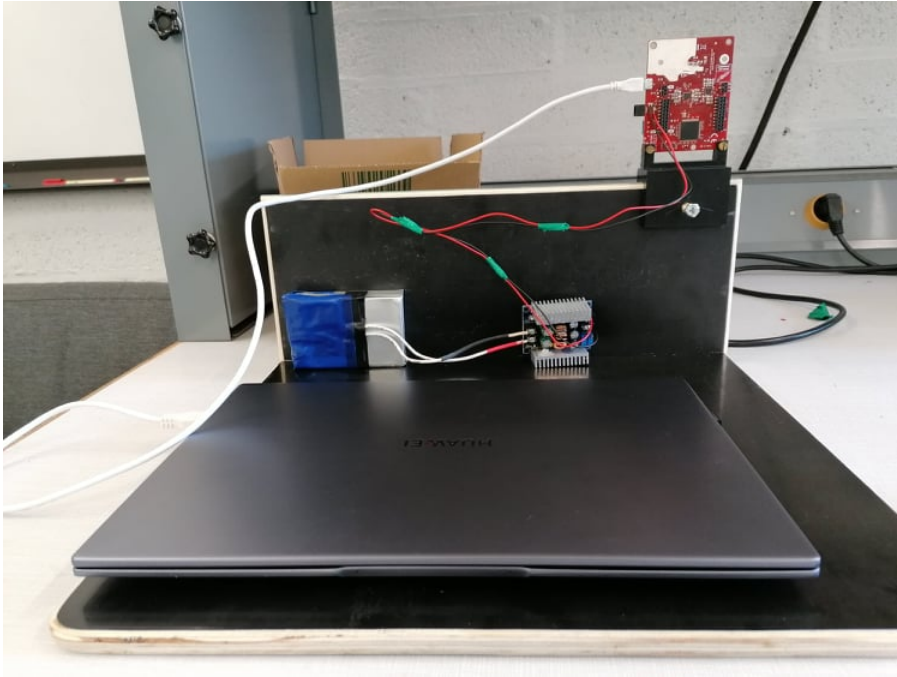


Figure A.1: Complete setup along with all components fixed.

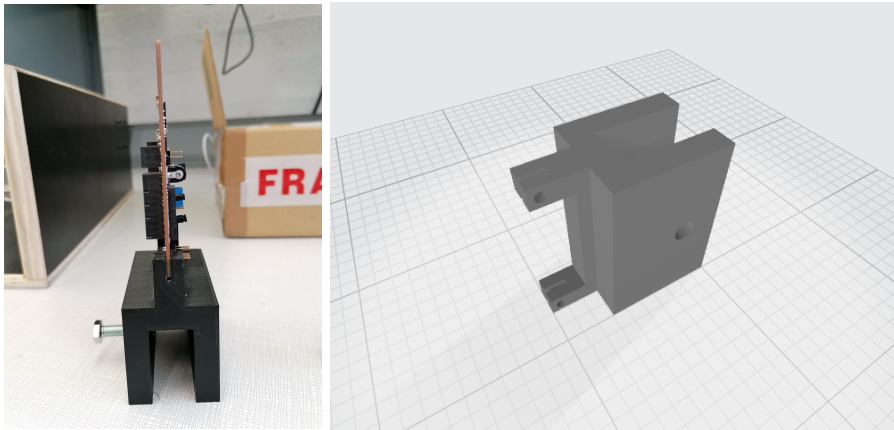


Figure A.2: Printed PCB radar holder.

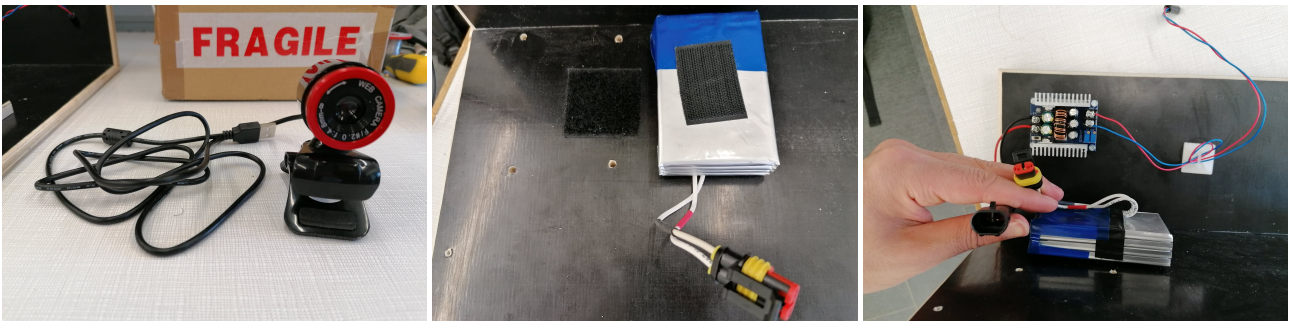


Figure A.3: Web camera to film test session.

Figure A.4: Battery adhesive to the wooden floor.

Figure A.5: Battery LiPo and converter fixed to the setup.

# Appendix B

## profile125

### B.1 Profile 125m

Tables B.1(a), B.1(b) and B.2 represent profile, chirp and frame parameters of LRR125m profile respectively.

Profile ID	0
Sampling rate	5000 Samp/s
Number of ADC samples	256
Start frequency	77 GHz
Frequency slope	5.33 MHz/ $\mu$ s
Idle time	5 $\mu$ s
Ramp end time	56 $\mu$ s
TX start time	1 $\mu$ s
ADC start time	4.8 $\mu$ s

((a)) SRR-125m Profile parameters.

	Chirp 0	Chirp 1
Profile ID	0	0
Start index	0	64
End index	63	127
Start frequency	0	0
Frequency slope	0	0
Idle time	0	11.8
ADC start time	0	0
TX antenna	1	1

((b)) SRR-125m Chirp parameters.

Table B.1: SRR-125m profile and chirp parameters.

Start index	0
End index	127
Loop count	1
Periodicity	30 ms
Number of real ADC samples	$256 \times 2 = 512$
Number of complex ADC samples	256
Chirp 0 number	$1 \times 64 = 64$
Chirp 1 number	$1 \times 64 = 64$
Total chirp number	$1 \times 128 = 128$
Transmitters number	1
Number of angle bins	32

Table B.2: LRR-125m Sub-frame parameters



### B.1.1 Derived parameters - SRR-125m

- **Maximum detected range:**

$$Range_{Max} = \frac{0.9 \times 5000 \times 10^6 (Samp/s) \times 3 \times 10^8 (m/s)}{2000 \times \frac{5.33 \times 10^6 Hz}{10^{-6} s}}$$

$$Range_{Max} = 126 \text{ m}$$

- **Range resolution:**

$$Range_{Res} = \frac{c \times SamplingRate}{2000 \times Slope \times ADCsamples}$$

$$Range_{Res} = \frac{3 \times 10^8 (m/s) \times 5000 \times 10^6 Hz}{2000 \times \frac{5.33 \times 10^6 Hz}{10^{-6} s} \times 256}$$

$$Range_{Res} = 0.55 \text{ m} = 55 \text{ cm}$$

- **Repetition period for chirp 0 and chirp 1 :**

$$Rep_{chirp} = ChirpIdleTime + ProfileIdleTime + RampEndTime$$

$$Rep_{Chirp_0} = 0 + 5 + 56 = 61 \text{ } \mu s$$

$$Rep_{Chirp_1} = 11.8 + 5 + 56 = 72.8 \text{ } \mu s$$

- **Speed resolution:**

$$Number_{Chirp_0} = (63 - 0 + 1) \times 1 = 64$$

$$Number_{Chirp_1} = (127 - 64 + 1) \times 1 = 64$$

$$Speed_{ResChirp_0} = \frac{1000}{61 \mu s} \times \frac{1}{64} \times \frac{3.9 mm}{2}$$

$$Speed_{ResChirp_0} = 0.49 \text{ m/s} = 1.76 \text{ Km/h}$$

$$Speed_{ResChirp_1} = 0.41 \text{ m/s} = 1.47 \text{ Km/h}$$

- **Maximum speed:**

$$Speed_{MaxChirp_0} = \frac{Speed_{ResChirp_0} \times Number_{Chirp_0}}{2}$$

$$Speed_{MaxChirp_0} = \frac{0.49 m/s \times 64}{2}$$

$$Speed_{MaxChirp_0} = 15.7 \text{ m/s} = 56.5 \text{ Km/h}$$

$$Speed_{MaxChirp_1} = 13.1 \text{ m/s} = 47.1 \text{ Km/h}$$

Code 2 and code 3 represent configuration code and the corresponding scene parameters of profile of LRR125m respectively.

```

// Initialize profile
#define PROFILE_ID (0U)
#define HPF1_VAL RL_RX_HPF1_175_KHz
4 #define HPF2_VAL RL_RX_HPF2_350_KHz
#define RX_GAIN_VAL (44U)
#define SAMPLERATE_VAL (5000U)
#define ADC_SAMPLE_VAL (256U)
#define IDLE_TIME_VAL (500U) // 5 us
9 #define RAMP_END_TIME_VAL (5600U)
#define START_FREQ_GHZ (77.0 f)
#define POWER_BACKOFF (0U)
#define TXPHASESHIFTER_VAL (0U)
#define FREQ_SLOPE_MHZ_PER_US (5.33 f)
14 #define TX_START_TIME_VAL (100U) // 1us
#define ADC_START_TIME_VAL (480U) // 4.8 us

// Initialize Chirp0
#define PROFILE_ID (0U)
19 #define START_INDEX_0 (0U)
#define END_INDEX_0 (63U)
#define START_FREQ_VAR_VAL_0 (0U) // Freq. variation value
#define FREQ_SLOPE_VAR_VAL_0 (0U)
#define IDLE_TIME_VAR_VAL_0 (0U)
24 #define ADC_START_TIME_VAR_VAL_0 (0U)
#define TX_CHANNEL_0 (TX_CHANNEL_1_ENABLE)

// Initialize Chirp1
#define PROFILE_ID (0U)
29 #define START_INDEX_1 (64U)
#define END_INDEX_1 (127U)
#define START_FREQ_VAR_VAL_1 (0U)
#define FREQ_SLOPE_VAR_VAL_1 (0U)
#define IDLE_TIME_VAR_VAL_1 (0U)
34 #define ADC_START_TIME_VAR_VAL_1 (0U)
#define TX_CHANNEL_1 (TX_CHANNEL_1_ENABLE)

// Initialize Frame
#define START_IDX (START_INDEX_0)
39 #define END_IDX (END_INDEX_1)
#define LOOP_COUNT (1U)
#define PERIODICITY_VAL (6000000U)
#define TRIGGER_DELAY_VAL (0U)
#define NUM_REAL_ADC_SAMPLES (ADC_SAMPLE_VAL * 2)
44 #define NUM_CMPLX_ADC_SAMPLES (ADC_SAMPLE_VAL)
#define NUM_CHIRP0 ((END_INDEX_0 - START_INDEX_0 + 1)*LOOP_COUNT)
#define NUM_CHIRP1 ((END_INDEX_1 - START_INDEX_1 + 1)*LOOP_COUNT)
#define NUM_TX (1U) // Only one TX is enabled
#define NUM_VIRT_ANT (NUM_TX * 4) // 4 is num. of RX
49 #define NUM_ANGLE_BINS (32U)
#define NUM_CHIRP_TOTAL ((END_IDX_1 - START_IDX_0 + 1) * LOOP_COUNT)

```

Listing 2: Configuration code of LRR125m profile.

```

// Deriving parameters
#define SPEED_OF_LIGHT      (300000000U)
#define LAMBDA_MILLIMETER  (SPEED_OF_LIGHT/START_FREQ_GHZ)

5 // Range resolution (m)
#define RANGE_RES          ((SPEED_OF_LIGHT * SAMPLERATE_VAL)/
                           (2000.0 f * FREQ_SLOPE_MHZ_PER_US *
                            NUM_CMPLX_ADC_SAMPLES))

10 // Repetition period (us)
#define CHIRP_0_REP_PERIOD (((IDLE_TIME_VAR_VAL_0 + IDLE_TIME_VAL
                              + RAMP_END_TIME_VAL)/100.0 f))
//divided by 100 to get it in us
#define CHIRP_1_REP_PERIOD (((IDLE_TIME_VAR_VAL_1 + IDLE_TIME_VAL
                              + RAMP_END_TIME_VAL)/100.0 f))

15 // Velocity resolution and maximum velocity of chirp 0 (m/s)
#define CHIRP_0_VEL_RES   (((1000.0 f/(CHIRP_0_REP_PERIOD))/
                            NUM_CHIRP_0)*(LAMBDA_MILLIMETER/2.0 f))
20 #define CHIRP_0_MAX_VEL (CHIRP_0_VEL_RES * NUM_CHIRP_0/2)

// Velocity resolution and maximum velocity of chirp 1 (m/s)
#define CHIRP_1_VEL_RES   (((1000.0 f/(CHIRP_1_REP_PERIOD))/
                            NUM_CHIRP_1)*(LAMBDA_MILLIMETER/2.0 f))
25 #define CHIRP_1_MAX_VEL (CHIRP_1_VEL_RES * NUM_CHIRP_1/2)

#define MIN_SNR_dB        (12.0 f) // minimum SNR value
#define NUM_CHIRPTYPES    (2U)

```

Listing 3: Scene parameters derivation of LRR125m profile.

# Appendix C

## Data set

In this appendix, a portion of the data set of location 3 experiment when the car was moving at speed of 30 Km/h, 50 Km/h, and 70 Km/h respectively will be presented, table C.1 shows the data set. However, it was chosen only one experiment to add a part of its data set and not the whole of it since it is a quiet large table. The data that resulted in the real time of the test during the corresponding transmitted frames are presented in the table, it was chosen only some frames just for illustration purposes and all other experiments are exactly same procedure. In fact, the sequence of these frames covers the whole trip of the car in both directions (go ↑ and back ↓) for the previous speed values, at a given frame it can be extracted the following info: the 3D position of the tracked object, its speed and noise and signal values.

The first column represents the number of the actual transmitted frame in the real time, as was explained in chapter 4, radar transmits an infinite number of frames each time is operated, the second column is the number of the detected objects at the corresponding frame, next three columns represent the Cartesian coordinates of the detected objects at the actual frame, the sixth column is the speed of the detected object, and last two columns represent signal to noise ratio and noise of the detected object. It can be pointed to figure 5.24 in chapter 5 to understand the data that are presented in the table C.1.

The data set was obtained using a Python script parser function should was inspired by the SDK of the device, when invoking the function on the output file(.dat file) that was obtained from the visializer, the result is basically a large excel or csv file. The parser\_mmw\_demo script outputs the detected point cloud data in mmw\_demo\_output.csv to showcase how to use the output of parser\_one\_mmw\_demo\_output\_packet. Python script parser function is shown in code list 4. The code of parser\_mmw\_demo script is not included here, however, only the demo version of parser function is included in code list 4.

```

import os
import sys
# import the parser function
from parser_mmw_demo import parser_one_mmw_demo_output_packet

# INPUT CONFIGURATION
# get the captured file name obtained from Visualizer
if (len(sys.argv) > 1):
    capturedFileName=sys.argv[1]
else:
    print ("Error: provide file name")
    exit()

# USE parser_mmw_demo SCRIPT TO PARSE ABOVE INPUT FILES
# Read the entire file
fp = open(capturedFileName, 'rb')
readNumBytes = os.path.getsize(capturedFileName)
print("readNumBytes: ", readNumBytes)
allBinData = fp.read()
print("allBinData: ", allBinData[0] ... allBinData[3])
fp.close()

# init local variables
totalBytesParsed = 0;
numFramesParsed = 0;

# Parser function extracts only one frame at a time
# so call this in a loop till end of file
while (totalBytesParsed < readNumBytes):

# Prints the parsed data to arrays
parser_result, \
headerStartIndex, \
totalPacketNumBytes, \
numDetObj, \
...
detectedNoise_array = parser_one_mmw_demo_output_packet
(allBinData[totalBytesParsed::1], readNumBytes-totalBytesParsed)

# Check the parser result
print ("Parser result: ", parser_result)
if (parser_result == 0):
totalBytesParsed += (headerStartIndex+totalPacketNumBytes)
numFramesParsed+=1
print("totalBytesParsed: ", totalBytesParsed)

```

Listing 4: Parser function script.

#Frame	#DetObj	X(m)	Y(m)	Z(m)	V(m/s)	SNR(0.1dB)	Noise(0.1dB)
3	0	0.000	0.000	0.000	0.000	124	323
4	0	0.000	0.000	0.000	0.000	135	310
11	1	-3.456	19.671	0.000	7.568	267	567
12	1	-3.865	19.986	0.000	7.911	270	543
20	1	-4.134	28.521	0.000	8.123	276	540
22	1	-3.913	29.111	0.000	8.233	271	551
34	0	0.000	0.000	0.000	0.000	112	301
35	1	-4.567	34.215	0.000	8.301	280	541
49	1	-3.766	71.875	0.000	8.333	298	530
59	1	-3.999	80.098	0.000	8.084	290	551
71	0	0.000	0.000	0.000	0.000	110	289
98	1	-4.097	90.318	0.000	-8.109	302	570
100	1	-4.432	88.480	0.000	-8.361	297	577
101	0	0.000	0.000	0.000	0.000	133	276
167	2	-4.4,-0.1	69.1,10.9	0.0,0.0	-8.3,0.4	297,50	541,101
201	2	-4.5,-0.15	25.1,8.9	0.0,0.0	-7.8,0.1	245,47	270,66
234	0	0.000	0.000	0.000	0.000	110	299
235	0	0.000	0.000	0.000	0.000	111	299
264	1	-5.169	21.814	0.000	-13.210	266	513
266	1	-5.220	21.990	0.000	-13.211	267	505
270	1	-5.719	27.017	0.000	-13.298	270	501
273	0	0.000	0.000	0.000	0.000	110	289
288	1	-9.861	46.717	0.000	-13.011	271	543
296	1	-11.345	69.091	0.000	-12.888	287	512
307	0	0.000	0.000	0.000	0.000	112	285
311	1	-24.633	112.898	0.000	-13.234	290	666
313	1	-24.712	113.008	0.000	-13.245	291	666
316	0	0.000	0.000	0.000	0.000	115	272
339	0	0.000	0.000	0.000	0.000	111	269
342	1	-23.118	112.871	0.000	-13.241	290	664
350	1	-19.101	81.096	0.000	-13.029	280	670
359	1	-14.520	60.313	0.000	-13.300	291	654
362	0	0.000	0.000	0.000	0.000	152	239
381	2	-6.5,-0.2	20.5,8.1	0.0,0.0	-12.8,0.8	291,48	520,98
384	2	-3.9,-0.2	9.5,7.6	0.0,0.0	-12.6,0.1	299,52	510,98
386	2	-2.6,-0.1	3.1,2.6	0.0,0.0	-11.1,-0.4	300,54	507,91
396	0	0.000	0.000	0.000	0.000	160	231
409	1	-11.515	17.060	0.000	-7.901	257	511
421	1	-18.569	63.413	0.000	-7.548	250	533
434	1	-26.376	84.404	0.000	-7.619	254	576
437	0	0.000	0.000	0.000	0.000	174	251
449	0	0.000	0.000	0.000	0.000	176	246
458	1	-12.601	63.140	0.000	-7.418	220	589
465	2	-7.1,-0.3	13.5,6.6	0.0,0.0	-7.6,0.2	276,51	501,88
470	1	-3.091	5.262	0.000	-7.519	241	576
472	0	0.000	0.000	0.000	0.000	170	233

Table C.1: A portion of the obtained data set during location 3 test



# Appendix D

## Technical issues

1. It was not easy at first to know how to run the device where there were some issues when trying to perform labs for different reasons such as incompatible versions for SDK, CCS, or other tools, and sometimes not proper device flashing process. That was a bit annoying because each lab has particular tools and versions, therefore it must download and install so many tools.
2. Another issue was when trying to run the lab under development mode using CCS where it was not able to find the right SDK, some time was needed to know where did the problem come from. It turns out later that CCS was not really seeing the right version of SDK or the other required tools in its directory therefore all lab files, different SDK versions (corresponding to each lab), CCS, or even any TI tool must be in the same directory to avoid any potential directory issues and, we had to refresh CCS library and install the new founded files with right versions. So at this moment, CCS can find everything it needs when executing any lab.
3. During configuring the device using PuTTY terminal many difficulties and error messages were encountered. First, when entering commands one by one, it was very difficult since there are 30 commands and each of which has its own arguments, therefore it was very rare not to make mistakes and to re-do the configuration process again and again, entering commands manually was a very bad idea in fact. Then, it was discovered that we can use high level copy-paste using '**Ins**' and '**shift**' buttons in the keyboard to insert configuration commands one by one to the terminal smoothly and without making mistakes, moreover it was very efficient and time-saving way (30 minutes in the first way to 5 minutes in the second one for each profile!).
4. One of the biggest issues when using PuTTY terminal was getting invalid configuration, at that moment we had to comeback and check every single argument in every single configuration command, therefore it was quiet critical to tune the value of each argument. Unfortunately at the programming the device phase, many documentations and source codes had to be read and understood well to be able to make a valid profile, one of these documents was the SDK datasheet where it was explained how to configure the device but without any link to the source code, that was a bit ambiguity since error messages that were came out did not have much explanation online. One of the most common error was when violating equation 3.5 when configuring profile.
5. When using the Estimator tool to recommend some values for a given profile, most cases were not working for one reason or another, in the lucky case, some adaption could be done



to obtain a valid profile and would work properly, otherwise, some mathematical calculation must be considered to obtain valid values for the desired profile.

# Bibliography

- [1] R. Aljasmí, *FMCW-radar, Signal Processing and Parameter Estimation*. FOI, SWEDISH DEFENCE RESEARCH AGENCY, 2002.
- [2] T. P. C. S. S. R. V. V. VASILII SEMKIN, JAAKKO HAARLA and C. OESTGES, *Analyzing Radar Cross Section Signatures of Diverse Drone Models at mmWave Frequencies*. IEEE Access, 2020.
- [3] T. I. (TI), “Awr1642 ods evm.” <https://www.ti.com/product/AWR1642>, 2018.
- [4] A. Meta, *Signal Processing of FMCW Synthetic Aperture Radar Data*. Sapienza Università di Roma, 2006.
- [5] D. K. A. Prima Aditya, Erna Apriliani and K. Baihaqi, *Estimation of three-dimensional radar tracking using modified Extended Kalman Filter*. IOP Conf. Series: Journal of Physics, 2018.
- [6] T. Tang and C. Wu, *Design of new Frequency Modulated Continuous Wave (FMCW) target tracking radar with digital beamforming tracking*. DRDC-Ottawa Research Centre, 2019.
- [7] R. Tutorials, “Frequency-modulated continuous-wave radar (fmcw radar), continuous-wave radar (cw).” [www.radartutorial.eu](http://www.radartutorial.eu), 1998.
- [8] Z. Peng and C. Li, *Portable Microwave Radar Systems for Short-Range Localization and Life Tracking*. Multidisciplinary Digital Publishing Institute (MDPI), 2019.
- [9] F. B. Florian Hau and M. Vossiek, *The Degradation of Automotive Radar Sensor Signals Caused by Vehicle Vibrations and Other Nonlinear Movements*. Multidisciplinary Digital Publishing Institute (MDPI), 2020.
- [10] T. I. (TI), “Short range radar (srr) user’s guide, developer’s guide.” <https://dev.ti.com/tirex/explore>.
- [11] S. R. Xiangyu Gao<sup>1</sup>, Guanbin Xing<sup>1</sup> and H. Liu, *Experiments with mmWave Automotive Radar Test-bed*. [eess.SP], 2022.
- [12] E. T. Inc., “Simulating a frequency-modulated continuous-wave (fmcw) radar system, rf tutorials.” <http://www.emagtech.com/wiki>, 2016.
- [13] A. M. Abhilash Nagesh Kashyap, *Identifiable Radar Reflectors For Automotive Pedestrian Safety*. HALMSTAD UNIVERSITY, 2010.
- [14] T. Thayaparan and C. Wernik, *Noise Radar Technology Basics*. Defence RD Canada – Ottawa, Technical Memorandum, 2006.
- [15] . Security, “Radar drone detection.” <https://www.911security.com>, 2020.

- [16] J. Portegies Zwart, *Aircraft Recognition from Features Extracted from Measured and Simulated Radar Range Profiles*. UvA-DARE (Digital Academic Repository), 2003.
- [17] K. Filter, "Extended kalman filter(ekf), multidimensional ekf." <https://www.kalmanfilter.net>, 2022.